

UNIVERSITÁ DEGLI STUDI DI PISA



**FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E
NATURALI**

CORSO DI LAUREA SPECIALISTICA IN INFORMATICA

Tesi di Laurea:

ESTRAZIONE DI CONOSCENZA E REALTÀ VIRTUALE

Candidato:

Pasqualina Goddi

Relatori:

Prof. Franco Turini

Dott.ssa Chiara Renso

Controrelatore:

Prof.ssa Francesca Levi

ANNO ACCADEMICO 2004/2005

Ai miei genitori

Sommario

■	1. INTRODUZIONE	1
▲	1.1 ORGANIZZAZIONE DELLA TESI	6
■	2. DATA MINING	9
▲	2.1 IL KNOWLEDGE DISCOVERY IN DATABASES (KDD)	12
	2.1.1 KDD: DEFINIZIONE	12
	2.1.2 FASI DEL PROCESSO KDD	13
	2.1.3 ARCHITETTURA DEL KDD	15
▲	2.2 TIPI DI DATA MINING	17
▲	2.3 TECNICHE DI DATA MINING	18
	2.3.1 ANALISI ASSOCIATIVA	18
	2.3.2 CLASSIFICAZIONE	19
	2.3.3 CLUSTERING	20
	2.3.4 ANALISI DELLE SERIE TEMPORALI	20
	2.3.5 ANALISI DEGLI OUTLIERS	21
▲	2.4 MISURE DI INTERESSE	22
▲	2.5 DATA MINING, OLAP, STATISTICA E INTELLIGENZA ARTIFICIALE	23
▲	2.6 DATA MINING: APPLICAZIONI	24
■	3. CLUSTERING E CLASSIFICAZIONE	27
▲	3.1 CLUSTERING: CHE COS'È	27
▲	3.2 CLUSTERING: RAPPRESENTAZIONE DEI DATI	29
▲	3.3 CLUSTERING: DISSIMILARITÀ	30
	3.3.1 VARIABILI NUMERICHE	30
	3.3.2 VARIABILI BINARIE	32
	3.3.3 VARIABILI NOMINALI	34
	3.3.4 VARIABILI ORDINALI	34
	3.3.5 VARIABILI DI TIPO MISTO	35

▲	3.4	CLASSIFICAZIONE DEI PRINCIPALI METODI DI CLUSTERING	36
▲	3.5	METODI DI PARTIZIONAMENTO: K-MEANS	38
▲	3.6	METODO GERARCHICO	41
▲	3.7	CLUSTERING DI TRAIETTORIE	43
	3.7.1	<i>DISTANZA EUCLIDEA VS LCSS</i>	44
▲	3.8	CLASSIFICAZIONE: CHE COS'È	46
▲	3.9	CLASSIFICAZIONE: FASI PRELIMINARI	47
▲	3.10	PROPRIETÀ DI UN BUON ALGORITMO DI CLASSIFICAZIONE	48
▲	3.11	TECNICHE DI CLASSIFICAZIONE	50
	3.11.1	<i>ALBERI DI DECISIONE</i>	50
	3.11.2	<i>ALBERI DI DECISIONE: TREE PRUNING</i>	53
▲	3.12	ALBERI DI DECISIONE E REGOLE DI CLASSIFICAZIONE	54
■	4.	LA REALTÀ VIRTUALE	57
▲	4.1	CREAZIONE DI UN AMBIENTE VIRTUALE	58
▲	4.2	SISTEMI DI REALTÀ VIRTUALE: AMBIENTI IMMERSIVI	59
▲	4.3	PRESENCE	61
	4.3.1	<i>BREAKS IN PRESENCE</i>	62
	4.3.2	<i>PARAMETRI FISIOLÓGICI</i>	63
	4.3.3	<i>QUESTIONARI SUS E ITC-SOPI</i>	64
▲	4.4	INTERAZIONE IN REALTÀ VIRTUALE: UNA PANORAMICA GENERALE	65
	4.4.1	<i>STRUMENTI PER LA VISIONE TRIDIMENSIONALE</i>	65
	4.4.2	<i>SISTEMI DI NAVIGAZIONE E STRUMENTI PER LA MANIPOLAZIONE DEGLI OGGETTI</i>	68
▲	4.5	SISTEMI PER IL RILEVAMENTO DELLA POSIZIONE IN VR	69
▲	4.6	UN LINGUAGGIO PER COSTRUIRE AMBIENTI VIRTUALI: VRML	70
▲	4.7	REALTÀ VIRTUALE E COMPUTER GRAPHICS A CONFRONTO	72
▲	4.8	REALTÀ VIRTUALE: APPLICAZIONI	74
■	5.	CASE STUDY: EQUATOR	77
▲	5.1	EQUATOR	77
	5.1.1	<i>L'AMBIENTE VIRTUALE</i>	79

	5.1.2	AVATARS	80
	5.1.3	FASI DELL'ESPERIMENTO	81
	5.1.4	RISULTATI	83
■	6.	ANALISI	85
▲	6.1	I DATI	86
	6.1.1	DESCRIZIONE DELL'AMBIENTE VIRTUALE	86
	6.1.2	DATI ANAGRAFICI	87
	6.1.3	PARAMETRI FISIologici	89
	6.1.4	TRAIETTORIE	90
	6.1.5	PRESENCE	90
▲	6.2	ANALISI DEI DATI: FASI E TECNICHE	91
▲	6.3	STUDIO DELLE TRAIETTORIE: RICOSTRUZIONE	94
	6.3.1	RICOSTRUZIONE: ASPETTI GENERALI	94
	6.3.2	ORGANIZZAZIONE DEI DATI DI INPUT	95
	6.3.3	RICOSTRUZIONE DELLE TRAIETTORIE: ALGORITMO	97
▲	6.4	CLUSTERING	99
▲	6.5	CLUSTERING CON K-MEANS: TRAIETTORIE RAPPRESENTATIVE	102
	6.5.1	TRAIETTORIE RAPPRESENTATIVE: ASPETTI GENERALI	102
	6.5.2	RIALLINEAMENTO TEMPORALE DELLE TRAIETTORIE	103
	6.5.3	TRAIETTORIE RAPPRESENTATIVE: ALGORITMO	104
	6.5.4	RAPPRESENTAZIONE GRAFICA DELLE TRAIETTORIE MEDIE	105
▲	6.6	CLUSTERS: ESEMPI	107
▲	6.7	CLASSIFICAZIONE	108
	6.7.1	DESCRIZIONE DEL DATASET	109
	6.7.2	ATTRIBUTO TARGET	110
	6.7.3	TRATTAMENTO DEI PARAMETRI FISIologici	110
	6.7.4	TARGET CLUSTER: STRATEGIA CLUSTERNOTCLUSTER	111
	6.7.5	BIPs: VALORI E INTERVALLI	114
	6.7.6	RISULTATI INTERESSANTI	116
▲	6.8	OTTIMIZZAZIONI	119
	6.8.1	ELIMINAZIONE ATTRIBUTI	119
	6.8.2	NORMALIZZAZIONE PARAMETRI FISIologici	120
	6.8.3	RISULTATI INTERESSANTI	122

▲	6.9 CLUSTERING: APPROCCIO ALTERNATIVO	126
	6.9.1 <i>ALGORITMO DI CLUSTERING: CALCOLO SIMILARITA`</i>	128
	6.9.2 <i>CLUSTERING: NORMALIZZAZIONE DEI DATI</i>	130
	6.9.3 <i>CLUSTERING: ORGANIZZAZIONE DEL DATASET DI INPUT</i>	131
	6.9.4 <i>CREAZIONE DEL FILE DI INPUT PER IL CLUSTERING: ALGORITMO</i>	132
	6.9.5 <i>RISULTATI INTERESSANTI</i>	133
■	7. CONCLUSIONI	135
■	APPENDICE A	137
■	APPENDICE B	149
■	APPENDICE C	223
■	APPENDICE D	255
■	BIBLIOGRAFIA	287

1. Introduzione

Negli ultimi anni si è registrata una crescita esplosiva nella capacità sia di generare che di collezionare dati in vari settori del mondo industriale e scientifico. Di fronte ad un evento di questo tipo si è reso necessario sfruttare al meglio questa nuova potenziale risorsa; i dati, infatti, se analizzati in maniera corretta consentono l'estrazione di informazioni interessanti e talvolta estremamente utili. Infatti l'informazione, in qualunque ambito, rappresenta conoscenza ed un'organizzazione che ne dispone può utilizzarla per le decisioni strategiche.

I dati rappresentano un bene prezioso in qualsiasi ambito. Si pensi ad esempio ad un'azienda che gestisce una grande catena di supermercati; uno studio della popolazione residente in prossimità dei supermercati può evidenziare una massiccia presenza di cittadini di origine asiatica. Di fronte ad un tale risultato l'organizzazione può decidere di riservare, in ogni supermercato, dei reparti specializzati in prodotti asiatici così da incentivare gli acquisti di questa parte della popolazione. In questo caso l'utilità dei dati acquisiti è fondamentale, soprattutto se la strategia adottata dai vertici decisionali dell'organizzazione induce un incremento delle vendite.

Se da un lato, la disponibilità di tanti dati è di primaria importanza, dall'altro pone una serie di problemi. Tra questi, uno tra i più importanti riguarda la scelta di strumenti idonei per elaborarli e per favorire l'estrazione di conoscenza. L'utilizzo di strumenti non adeguati porta ad un'analisi poco proficua e talvolta fuorviante. L'analisi dei dati richiede uno studio approfondito del tipo di informazione che si vuole estrarre e degli strumenti più idonei per trattarli. La scelta degli strumenti per l'estrazione di conoscenza è di primaria importanza. L'importanza degli strumenti in questo contesto è paragonabile a quella degli

strumenti da impiegare per la realizzazione di un circuito elettronico. La realizzazione del circuito richiede la definizione di diversi aspetti. In fase di progettazione è necessario prima di tutto individuare i componenti da utilizzare per la sua realizzazione. Questa fase deve essere realizzata con particolare cura in quanto richiede uno studio di compatibilità tra i diversi componenti.

Terminata questa fase, è necessario individuare gli strumenti da impiegare per l'assemblaggio. L'uso di strumenti non idonei può essere rischioso e può compromettere le funzionalità dei vari componenti rendendoli inutilizzabili per la realizzazione del circuito. I dati presenti in un'organizzazione sono paragonabili ai componenti di un circuito. L'utilizzo di strumenti inadeguati può portare all'individuazione di informazioni poco interessanti. In alcuni casi, quando gli strumenti prevedono una rielaborazione dei dati, si può avere una modifica irreversibile di questi ultimi che può compromettere le analisi successive.

Il problema del trattamento dei dati per l'estrazione di conoscenza diventa sempre più serio, soprattutto se le organizzazioni ne dispongono in grosse quantità. In questo caso gli strumenti di analisi tradizionali, soprattutto di tipo statistico, risultano inadeguati: si tratta infatti di strumenti che offrono buone prestazioni quando applicati su piccoli campioni; la loro efficacia va via via attenuandosi all'aumentare del numero di dati su cui vengono applicati. E' proprio in questo contesto che si inserisce il *data mining*, inteso come un insieme di strumenti di modellistica statistica volto a scoprire relazioni nascoste nella miriade di dati di una determinata organizzazione.

Il data mining si inserisce nel più vasto campo del *Knowledge Discovery in Databases* (KDD). Talvolta questi due termini sono utilizzati come sinonimo uno dell'altro, ma sarebbe più preciso parlare di *Knowledge Discovery* per riferirsi al processo di estrazione della conoscenza e di data mining come una particolare fase del KDD, ossia l'applicazione di uno specifico algoritmo per l'individuazione di associazioni, "patterns", sequenze ripetute, regolarità nascoste nei dati. Per capire meglio lo scopo del data mining si consideri il

significato di questo termine: il verbo *to mine* in inglese indica proprio l'attività tipica delle miniere in cui si scava in profondità per estrarre minerali nascosti e preziosi. Associandolo al termine *data* si può allora intuire l'analogia per cui, quando ci si trova di fronte ad una grande mole di dati si deve scavare a fondo per estrarne informazioni nascoste o di difficile accesso. L'importanza dei dati come depositari di conoscenza è nota in tutti i campi e le tecniche di data mining trovano applicazione in diversi settori, tra i quali l'economia, le telecomunicazioni, la finanza, la biologia.

L'idea di questo lavoro di tesi è nata dalla disponibilità di una certa quantità di dati relativi ad un settore nuovo nel quale il data mining non ha ancora trovato applicazione: la *realtà virtuale*. Si tratta di una disciplina che consente la creazione di un ambiente artificiale in cui un utente può muoversi e manipolare oggetti virtuali.

Gli esperimenti di realtà virtuale sono condotti per diversi scopi; ad esempio, in ambito militare, per l'addestramento dei piloti o anche in ambito medico per la pianificazione di un intervento chirurgico particolarmente delicato. Oggetto di ulteriori esperimenti di realtà virtuale è lo studio dell'interazione tra uomo e mondo virtuale. A prescindere dalla tipologia e l'ambito in cui si colloca, un esperimento di realtà virtuale permette l'estrazione di una grossa quantità di dati. Questi possono rappresentare una fonte informativa preziosa così come avviene in altri settori. Si consideri ad esempio l'impiego della realtà virtuale in campo militare: attraverso un'analisi dei parametri fisiologici dei soggetti sottoposti ad addestramento è possibile individuare gruppi di soggetti che hanno le stesse reazioni emotive. Uno studio di questo tipo può essere utile in una fase successiva in cui i militari devono confrontarsi con la realtà. Sulla base dei gruppi individuati precedentemente, si può decidere per le prime missioni, di suddividere quelli che risultavano più stressati in gruppi diversi. L'organizzazione dei militari che risultano più stressati nello stesso gruppo potrebbe compromettere infatti la buona riuscita della missione.

Se un'applicazione di tecniche di data mining fosse possibile se ne trarrebbero grossi vantaggi, di natura diversa a seconda dell'esperimento condotto. Il data mining potrebbe essere utilizzato come supporto alle altre tecniche di analisi utilizzate su dati virtuali, prevalentemente di tipo statistico. L'esito positivo di un'applicazione di questo tipo, oltre a portare a dei vantaggi nel campo della realtà virtuale, rappresenterebbe una grossa conquista per il data mining: esisterebbe un nuovo campo applicativo per questo potente strumento. L'obiettivo di questo lavoro è verificare se una tale applicazione è possibile e definirne una possibile metodologia. L'approccio migliore per un studio di questo tipo è partire da un esperimento di realtà virtuale realmente condotto e considerare i dati estratti da un tale esperimento. La verifica consiste nell'applicare le tecniche di data mining più idonee su questi dati e verificare la validità dei risultati.

I dati estraibili da un esperimento virtuale risultano diversi da situazione a situazione e dipendono dal problema specifico per il quale è stato condotto. Uno degli elementi che contraddistingue due esperimenti diversi è l'ambiente ricreato.

L'esperimento dal quale inizia questo lungo studio è stato condotto presso il dipartimento di informatica della *University College London*. Si tratta della ricostruzione di uno shopping center popolato da quelli che nel linguaggio "virtuale" vengono chiamati *avatars*. L'esperimento oggetto del nostro studio ha coinvolto 40 partecipanti, un gruppo abbastanza eterogeneo in termini di età, occupazione ed altri aspetti.

Lo scopo specifico per il quale è stato condotto l'esperimento è lo studio dell'interazione tra uomo e mondo virtuale e sue reazioni fisiologiche. Questo esperimento è inserito in un contesto molto più ampio in cui si studia il livello di *presenza* di un soggetto in un ambiente virtuale. Un soggetto è presente nel mondo virtuale nella misura in cui risponde agli stimoli che provengono da questo. Lo studio di questo fenomeno richiede l'analisi dei cosiddetti *breaks in*

presence (BIPs). Si ha un break in presence quando l'utente sente gli stimoli provenienti dalla realtà prevalere su quelli dell'ambiente virtuale: è cosciente dell'irrealisticità dell'esperienza che vive. Tale esperimento ha prodotto una serie di dati eterogenei riguardanti le traiettorie seguite dagli individui, i parametri fisiologici e i dati anagrafici. La nostra analisi ha richiesto l'esecuzione di diverse fasi. La prima di queste fasi riguarda la scelta dei dati su cui applicare le tecniche di data mining. Questa selezione ha portato ad individuare come dati interessanti, oltre ai dati sopraindicati, i breaks in presence e le caratteristiche dell'ambiente virtuale.

La seconda fase della nostra analisi consiste nella definizione degli obiettivi, ovvero dei tipi di informazione che si vogliono estrarre, compatibili naturalmente con il tipo di dati disponibili. Uno studio scrupoloso di questi dati pone in evidenza come da un lato, le informazioni derivabili possono fornire un valido contributo a problemi in fase di studio, dall'altro possono fornire conoscenza completamente nuova.

Nel primo caso ad esempio, poiché uno dei temi caldi del settore è la relazione tra il livello di presenza e la realistica dell'ambiente virtuale, può essere interessante utilizzare il data mining per cercare un legame tra i BIPs e le caratteristiche dell'ambiente. Strumenti di questo settore ci consentono anche di cercare delle relazioni tra i BIPs e altri dati come ad esempio età o sesso. Risultati di questo tipo possono arricchire lo studio degli esperti di realtà virtuale.

Nel secondo caso, le tecniche di data mining possono essere utilizzate per estrarre conoscenza che gli strumenti di analisi tradizionale non consentono di individuare. In questo caso la nostra scelta è stata quella di condurre uno studio centrato sulle traiettorie compiute dai partecipanti all'esperimento. Si è ritenuto potesse essere interessante individuare gruppi di soggetti aventi una traiettoria simile e individuare all'interno di ciascun gruppo similarità definita in termini di altri dati.

La definizione degli obiettivi ha rivestito un ruolo cruciale nella scelta degli strumenti da impiegare nell'analisi. Gli strumenti di data mining ritenuti più idonei sono il clustering e la classificazione.

Per quanto riguarda il clustering, sono stati sfruttati i risultati già raggiunti nello studio delle traiettorie, in particolare sul clustering di traiettorie. Gli algoritmi di clustering tradizionali operano su entità che hanno una collocazione fissa nello spazio, mentre le traiettorie sono relative a entità che si muovono nello spazio e che ad istanti temporali diversi si trovano in un punto diverso. Il clustering di traiettorie rappresenta un aspetto importante del data mining perché ne consente l'applicazione in vari settori, per esempio in biologia è utilizzata per lo studio del processo migratorio degli animali.

La classificazione è risultata utile nella descrizione di ciascun cluster di traiettorie e nella definizione di un modello per i BIPs.

Le fasi successive sono state di applicazione di queste tecniche e valutazione dei risultati.

1.1 Organizzazione della tesi

Nel capitolo 2 si fornisce una breve introduzione al data mining con la descrizione delle principali tecniche e degli ambiti applicativi.

Il capitolo 3 descrive, in maniera più approfondita rispetto a quanto fatto nel capitolo 2, le tecniche utilizzate in questo lavoro di tesi: clustering e classificazione

Il capitolo 4 fornisce un'introduzione alla realtà virtuale con una descrizione dei dispositivi di visualizzazione, degli strumenti di navigazione e degli ambiti applicativi.

Il capitolo 5 contiene una descrizione dell'esperimento di realtà virtuale da cui sono stati estratti i dati oggetto della nostra analisi.

Il capitolo 6 è la descrizione delle varie fasi della nostra analisi e dei risultati più significativi.

Infine il capitolo 7 rappresenta una valutazione dell'analisi fatta e di quelli che potrebbero essere gli sviluppi futuri di questa integrazione del data mining nella realtà virtuale.

2. Data Mining

Negli ultimi anni si è avuta una crescita esponenziale di dati; di fronte all'aumento di questa nuova potenziale risorsa, è nata l'esigenza di sfruttarla al meglio per poterne estrarre informazione utile per ottenere vantaggi economici e non. I dati risultano utili nella misura in cui è possibile estrarre delle informazioni interessanti e l'importanza della loro scoperta cresce di pari passo alla criticità dei problemi che aiutano a risolvere.

Talvolta accade che l'analisi di un insieme di dati non porti ad ottenere informazioni e questo può essere dovuto a diversi fattori. Tra questi, uno potrebbe essere il fatto che i dati non contengano conoscenza e non risultano utili in alcun modo, indipendentemente dalle tecniche di elaborazione utilizzate. Un'altra causa della mancata estrazione di conoscenza è l'utilizzo di metodi non adatti per trattare il tipo di dato specifico. E' importante quindi utilizzare metodi alternativi più idonei.

L'utilizzo di metodi non efficaci può far sì che il potere informativo di un insieme di dati possa essere sfruttato solo in minima parte e questo può danneggiare l'analisi nel caso in cui l'analista non si renda conto che la povertà di conoscenza associata ai dati non è intrinseca ad essi, ma è dovuta ad un'analisi sbagliata. La scelta del metodo di analisi non è facile, in letteratura ne esistono molti, ciascuno con le proprie caratteristiche. Alcuni metodi di analisi offrono buoni risultati con dati di un certo tipo o con una certa quantità di dati.

Un buon metodo di analisi di dati dovrebbe essere caratterizzato da una buona *scalabilità*, nel senso che le prestazioni offerte non cambiano al variare della quantità di dati che si analizzano; un buon metodo dovrebbe essere in grado di trattare dati eterogenei e ridondanti. In quest'ultimo caso quindi dovrebbe

essere capace di individuare dati di cui esistono diverse rappresentazioni e darne un'unica rappresentazione. Nel caso in cui si debba trattare una grossa mole di dati, i metodi di analisi tradizionale risultano essere inadeguati. Gli strumenti di analisi tradizionale si suddividono in strumenti di *analisi statistica* e strumenti tipici di interrogazioni di dati (*data retrieval*). Per quanto riguarda i primi, le difficoltà dipendono da diversi fattori. Un primo fattore è che difficilmente operano su grandi quantità di dati, infatti richiedono operazioni di campionamento con conseguente perdita di informazioni.

Inoltre spesso richiedono valori di tipo quantitativo, mentre i dati talvolta sono di tipo qualitativo. Infine, non gestiscono i valori mancanti e richiedono personale tecnico sia per il loro utilizzo che per l'interpretazione dei risultati.

Per quando riguarda il data retrieval, le difficoltà principali riguardano i tempi di risposta che aumentano all'aumentare della quantità di dati e l'inadeguatezza nell'individuare associazioni nascoste. Il data retrieval è infatti uno strumento per interrogare banche dati e consiste nel formulare una query o interrogazione. Il sistema cerca all'interno della banca dati tutti i casi che soddisfano le condizioni nella query, (cioè tutti i dati che presentano le caratteristiche richieste) e fornisce la risposta. L'individuazione di "associazioni nascoste" può quindi solo procedere per tentativi.

Proprio in questo contesto si inserisce il *data mining* che si propone come soluzione per i problemi sopra elencati. Il data mining viene utilizzato per l'elaborazione di informazioni contenute nei dati in sostituzione o a complemento dei metodi di analisi tradizionali.

Il data mining può essere definito come "*l'estrazione automatica di informazioni predittive nascoste, da grandi masse di dati rendendole visibili*" [Hand01]. In questa definizione:

- Il termine *automatica* evidenzia la necessità di utilizzare mezzi informatici quali elaboratori che, grazie all'utilizzo di software adatti, compiono l'estrazione.

- Il termine *predittive* è posto ad evidenziare un fine dell'estrazione che è quello di poter prevedere, sulla base dei dati che stiamo analizzando, l'andamento futuro degli stessi.
- Il termine *nascoste*, si riferisce al fatto che chi cerca queste informazioni non ne conosce l'esistenza e non saprebbe come venirne a conoscenza se non con l'applicazione del data mining.

Essendo il data mining una disciplina di recente sviluppo, non ne esiste una sola definizione. Tra le tante esistenti, particolarmente interessanti sono le seguenti:

- *“Il data mining è l'esplorazione e l'analisi, attraverso mezzi automatici e semiautomatici, di grosse quantità di dati allo scopo di scoprire modelli e regole significative”* [Ber97].
- *“Il data mining si riferisce all'uso di una varietà di tecniche per identificare “pepite” di informazione e di conoscenza per il supporto al decision making. L'estrazione di tale conoscenza avviene in modo che essa possa essere usata in diverse aree come supporto alle decisioni, previsioni e stime. I dati sono spesso voluminosi, ma così come sono, hanno un basso valore e nessun uso diretto può esserne fatto; è l'informazione nascosta nei dati che è utile”* [Lar05].
- *“Il data mining è la non banale estrazione di informazione implicita, precedentemente sconosciuta e potenzialmente utile attraverso l'utilizzo di differenti approcci tecnici”* [Fra91].
- *“Il data mining consiste nell'uso di tecniche statistiche da utilizzare con i databases aziendali per scoprire modelli e relazioni che possono essere impiegati in un contesto di business”* [Man00].

2.1 Il *Knowledge Discovery in Database* (KDD)

Il data mining è la fase più importante di un processo, chiamato *Knowledge Discovery in Database* (KDD). Tale processo ha come obiettivo quello di estrarre conoscenza da banche dati di grandi dimensioni tramite l'applicazione di algoritmi che individuano le associazioni "nascoste" tra le informazioni e le rendono esplicite. Spesso il termine data mining viene utilizzato erroneamente per riferirsi all'intero processo KDD anche se ne costituisce solo una fase, in particolare quella in cui si applica un particolare algoritmo per individuare *associazioni, patterns, sequenze ripetute*, in generale regolarità nascoste nei dati. Il motivo di questo abuso di linguaggio è che la fase di data mining è di gran lunga quella più importante all'interno del KDD e in essa si ripetono molte delle fasi del KDD stesso.

2.1.1 KDD: definizione

Il termine Knowledge Discovery in Databases (KDD), si riferisce all'intero "processo di scoperta della conoscenza che consiste nell'identificazione di relazioni tra dati che siano valide, nuove, potenzialmente utili e comprensibili" [Lar05].

Alcuni concetti da mettere in evidenza nella definizione sono i seguenti:

- *Processo di scoperta della conoscenza*: un insieme di attività che consistono nella preparazione dei dati, la ricerca di relazioni, la valutazione e il raffinamento della conoscenza estratta.
- *Relazioni valide*: sono tali se valgono con un certo grado di certezza, anche su dati diversi da quelli usati per la scoperta delle stesse.
- *Relazioni nuove*: la novità può essere misurata rispetto ai cambiamenti nei dati (confrontando i valori correnti con quelli precedenti o quelli attesi) o nella conoscenza (cioè come una nuova scoperta è collegata a quella precedente).

- *Relazioni utili:* devono poter condurre a delle azioni utili. Per esempio la scoperta di una dipendenza fra articoli acquistati da uno stesso cliente in un supermercato potrebbe attivare opportune strategie di marketing.
- *Relazioni comprensibili:* per gli utenti, per facilitare una migliore comprensione dei dati coinvolti.

2.1.2 Fasi del Processo KDD

Il processo KDD si articola in diverse fasi:

- *Definizione degli obiettivi dell'utente finale.* E' la fase più critica dell'intero processo in quanto influenza le fasi successive; gli obiettivi vanno esplicitati in modo chiaro e senza lasciare spazio a dubbi e incertezze. Ha molto in comune con le fasi iniziali di un qualunque progetto commerciale, durante le quali si individuano gli obiettivi di business da raggiungere. In questa fase è importante la collaborazione che si stabilisce tra gli analisti del business e gli analisti dei dati, i quali devono comprendere il dominio applicativo ed estrarre la conoscenza già esistente.
- *Individuazione delle fonti dati.* L'obiettivo di questa fase è identificare le sorgenti di dati disponibili ed estrarne un campione per le analisi preliminari. Oltre alle variabili selezionate, è necessario acquisire le corrispondenti informazioni semantiche (metadati), indispensabili per capire il significato di ciascuna variabile. I metadati potrebbero includere la definizione dei dati, la descrizione dei tipi, i valori potenziali, il loro sistema sorgente e il formato. Questa fase è molto importante perché si selezionano i dati significativi, tralasciando dati e informazioni che possono risultare inutili. Si pensi per esempio allo studio delle associazioni tra i prodotti di una catena di supermercati. In questo caso non ha senso conservare i dati relativi alla professione dei clienti; è invece errato non considerare tale variabile, che

potrebbe fornire utili informazioni relative al comportamento di determinate fasce di clienti, nel caso in cui si voglia effettuare un'analisi discriminante.

- *Pre-elaborazione.* I dati selezionati devono essere pre-elaborati al fine di rimuovere eventuali inconsistenze, rimuovere o ridurre l'effetto rumore (*data cleaning*), eliminare casi limite (outlier), decidere le strategie per la gestione dei valori nulli, scartare dati obsoleti. Affinché i dati siano utilizzabili, talvolta è necessario trasformarli. Si possono convertire tipi di dati in altri o definire nuovi dati attraverso l'uso di operazioni matematiche e logiche sulle variabili. Inoltre quando i dati provengono da fonti diverse, è necessario effettuare una loro riconfigurazione al fine di garantire la consistenza. Nel caso ci siano più sorgenti dati, ognuna di esse ha la propria rappresentazione e organizzazione; può accadere che uno stesso dato sia contenuto in due sorgenti diverse con una rappresentazione diversa. In questo caso se il dato è interessante, può essere utile tener conto delle informazioni ad esso associate nelle varie sorgenti e darne una rappresentazione unica.
- *Data mining.* Consiste nell'analizzare i dati con particolari tecniche che consentono di estrarre patterns di dati. Ai dati trasformati vengono applicate una serie di tecniche in modo da poterne ricavare dell'informazione non banale o scontata. I tipi di dati che si hanno a disposizione e gli obiettivi che si vogliono raggiungere, possono dare un'indicazione circa il tipo di metodo/algoritmo da scegliere per la ricerca di informazioni dai dati.
- *Patterns evaluation.* Durante questa fase si valuta l'output della fase precedente. Il data mining crea dei patterns, ovvero dei modelli, che possono costituire un valido supporto alle decisioni. Non basta però interpretare i risultati attraverso dei grafici che visualizzano l'output del data mining, ma occorre valutare questi modelli e cioè capire in che misura questi possono essere utili. In particolare, si fa una selezione dei patterns interessanti estratti nella fase precedente, utilizzando misure di interesse.

Successivamente, alla luce di risultati non perfettamente soddisfacenti, è possibile rivedere una o più fasi dell'intero processo KDD.

- *Knowledge presentation*: in questa fase vengono utilizzate tecniche di visualizzazione e rappresentazione della conoscenza per mostrare all'utente i patterns estratti.

Il KDD è un processo iterativo, nel senso che uno o più fasi possono essere ripetute in maniera ciclica, come evidenziato in figura 2.1, fino a quando non si ottengono dei risultati soddisfacenti.

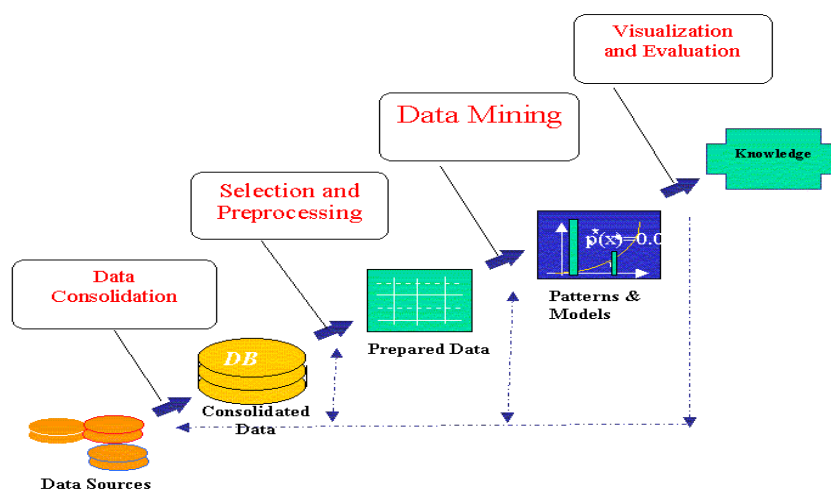


Figura 2.1 Fasi del KDD

2.1.3 Architettura del KDD

Un tipico sistema di data mining è costituito da un insieme di componenti, ciascuno dei quali assolve a certe funzionalità e interagisce con gli altri componenti.

Gli elementi più importanti in un sistema di data mining sono i dati, memorizzati in databases e data warehouse o altri contenitori. Nell'architettura di un sistema di data mining questi costituiscono un elemento indispensabile, senza il quale il processo non potrebbe essere realizzato, come mostrato in figura 2.2. Sui dati

organizzati in questi supporti si effettua la pulizia (*data cleaning*) e successivamente l'integrazione dei dati provenienti da fonti diverse (*data integration*). Queste operazioni vengono effettuate dal server del database o del data warehouse tenendo conto di quelle che sono state le richieste dell'utente. Tra le componenti del sistema, particolarmente importante è il *data mining engine* che realizza il mining vero e proprio e sottopone i dati ad algoritmo per realizzare classificazione, clustering, associazioni, analisi di deviazione. Una componente di supporto al data mining engine è costituito dalla base di conoscenza, utilizzata come guida per la ricerca e la valutazione dei patterns. Il data mining engine interagisce con un'altra componente, il modulo di valutazione dei patterns, che individua tra i patterns estratti con le varie tecniche di data mining, quelli più interessanti. Un'altra componente del KDD è il Graphical user Interface che crea il collegamento tra utente e sistema di data mining; in particolare, fornisce un'interfaccia attraverso la quale l'utente può formulare query e attraverso la quale viene visualizzato il risultato della richiesta, dopo le varie elaborazioni.

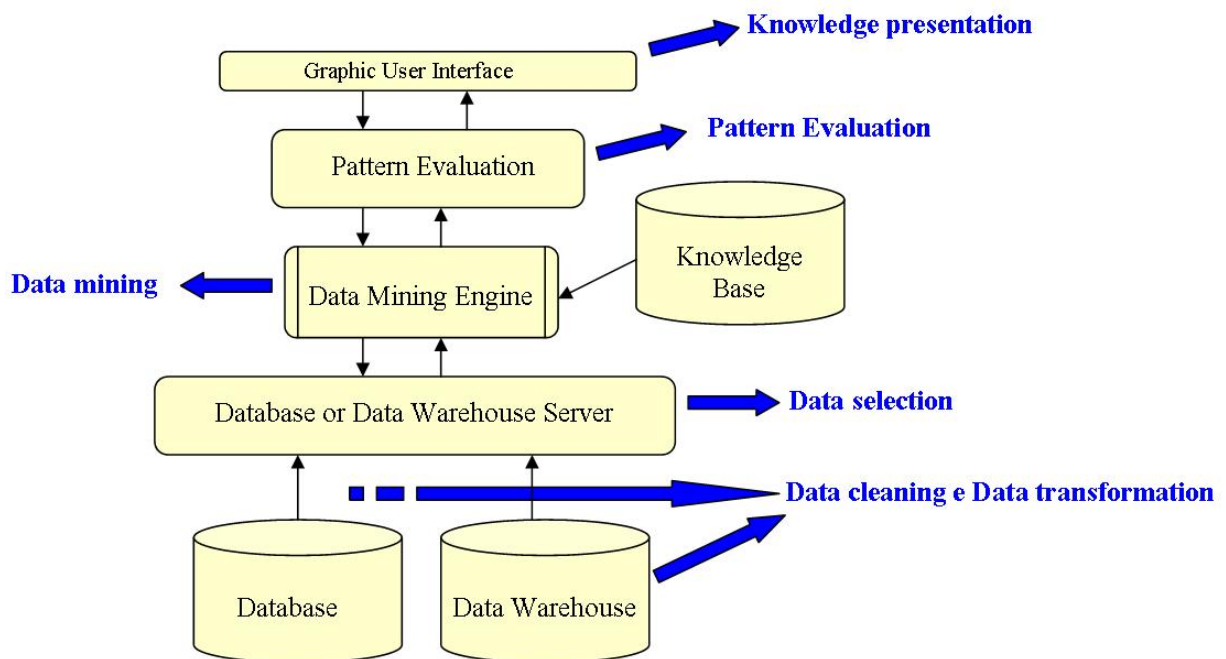


Figura 2.2 Architettura di un sistema KDD

2.2 Tipi di Data Mining

Gli approcci al data mining possono essere di due tipi: di tipo descrittivo e di tipo predittivo. Nel primo caso si tratta di utilizzare la statistica come guida per l'esplorazione dei dati, cercando di trovare conferme a fatti che l'utente ipotizza o già conosce, o per migliorare la comprensione di fenomeni parzialmente conosciuti. In questo ambito vengono utilizzate le statistiche di base che permettono di ottenere descrizioni brevi e concise del dataset, di evidenziare interessanti e generali proprietà dei dati; è anche possibile l'utilizzo di tecniche statistiche tradizionali quali, ad esempio la regressione. In un approccio di questo tipo riveste un ruolo cruciale l'utente, al quale è affidato il compito di formulare delle ipotesi su alcune variabili.

La sola descrizione dei dati non può fornire quelle informazioni di supporto alle decisioni, che contraddistinguono il data mining da altri metodi. In un approccio di tipo previsivo, l'utente analizza i dati alla ricerca di informazioni che a priori ignora. In questo caso, si costruiscono uno o più insiemi di modelli, si effettuano delle inferenze sui dati disponibili e si tenta di prevedere il comportamento di nuovi dataset.

Gli approcci predittivi costituiscono la parte più rilevante delle tecniche di data mining, risultano di gran lunga più interessanti. In questi tipi di modelli all'utente non è affidato nessun tipo di compito specifico, è il sistema che scopre "automaticamente" informazioni rilevanti nascoste nei dati: si cerca di individuare patterns frequenti, regole di associazione, valori ricorrenti.

2.3 Tecniche di data mining

Il data mining utilizza diverse tecniche di analisi dei dati e di conseguenza anche diversi algoritmi che le implementano. A seconda del problema che si deve affrontare, si sceglie il metodo che si reputa più idoneo. I principali metodi di data mining sono: l'analisi associativa, il clustering, la classificazione, l'analisi delle serie temporali e l'analisi degli outliers.

Di seguito, è riportata una breve descrizione di queste tecniche di analisi dei dati. Per una trattazione più dettagliata, si rimanda al capitolo 3 per la descrizione di clustering e classificazione, i metodi usati in questa tesi, mentre rimandiamo ai testi [Ber97, Fra91, Han00] per dettagli sugli altri metodi.

2.3.1 Analisi associativa

Questo tipo di analisi consente di individuare le regole di implicazione logica presenti nella base di dati, quindi di scoprire gruppi di affinità tra oggetti. Le regole di associazione che si determinano sono nella forma $X \Rightarrow Y$, dove X e Y contengono condizioni attributo-valore. La regola associativa $X \Rightarrow Y$ va interpretata come “le tuple di database che soddisfano le condizioni in X , soddisfano anche le condizioni in Y ”. Il campo di applicazione tipico delle regole associative è la *market-basket analysis*, ovvero l'analisi dei prodotti acquistati insieme dai clienti, volta a determinare le correlazioni più frequenti tra prodotti. Per esempio, da un'analisi degli acquisti fatti in una particolare catena di supermercati, può risultare che chi acquista il latte, acquista anche uova. Questa regola può essere rappresentata come “latte \Rightarrow uova”. Le regole estratte non sono valide in assoluto, ma valgono con una certa probabilità. In particolare per valutare se una regola sia interessante si utilizzano due misure, *supporto* e *confidenza*.

Data una regola del tipo $X \Rightarrow Y$, il *supporto* è definito come la percentuale di transazioni che contengono sia X che Y , cioè la probabilità $P(X \cup Y)$. La *confidenza* è la percentuale di transazioni, tra quelle che contengono X , che contengono anche Y , cioè $P(Y | X)$. Considerando la regola “latte \Rightarrow uova” con supporto= 77% e confidenza= 85%, allora si può dedurre che l’85% dei clienti che acquistano il latte acquistano anche le uova; un supporto pari al 77% sta ad indicare che l’abbinamento tra uova e latte viene riscontrato nel 77% delle transazioni considerate. Le regole associative possono essere unidimensionali se contengono un solo predicato (è il caso della regola data in precedenza), oppure multidimensionali se esiste un’associazione tra più attributi. Un esempio di regola multidimensionale, può essere “uova & pane \Rightarrow latte”, che indica il fatto che chi acquista congiuntamente uova e pane, acquista anche il latte. Tra le applicazioni delle regole associative, vanno citati lo studio delle abitudini di acquisto per la pubblicità mirata, per l’organizzazione della merce sugli scaffali e per lo studio della variabilità delle vendite in assenza di un certo prodotto. Nonostante le regole associative nascano nel campo del market basket analysis, trovano applicazione anche in altri campi applicativi; per maggiori dettagli si rimanda al paragrafo 2.6.

2.3.2 Classificazione

Lo scopo di questa tecnica di data mining è trovare un insieme di modelli o funzioni che descrivono e distinguono classi di dati. La definizione di questi modelli viene fatta prendendo in considerazione un insieme di oggetti già provvisti di *etichetta di classe*; sulla base dei valori degli attributi di questi oggetti si dà una descrizione generale di ogni classe, ovvero si cercano delle affinità nei valori degli attributi che caratterizzano gli oggetti che vi appartengono.

Una volta derivati, i modelli possono essere utilizzati per predire la classe di appartenenza di un particolare oggetto privo di etichetta di classe.

Ogni modello è definito da un insieme di regole che pongono dei vincoli sui valori degli attributi. Avendo a disposizione un modello per le varie classi e un oggetto x , quest'ultimo appartiene alla classe A se, i valori dei suoi attributi soddisfano le regole che definiscono la classe A .

Un'applicazione tipica della classificazione è la valutazione delle categorie di rischio dei clienti per le società che concedono mutui e prestiti.

2.3.3 Clustering

Lo scopo di questa tecnica di mining è quello di raggruppare oggetti in clusters o gruppi, in modo tale che oggetti appartenenti alla stessa classe siano molto simili e diversi da un qualunque oggetto appartenente ad altre classi. Il clustering si basa quindi su un principio che consiste nel massimizzare la similarità intraclasse e minimizzare la similarità interclasse. Poiché le categorie individuate non sono note a priori, spesso in riferimento a questa tecnica si parla anche di apprendimento senza supervisore. Tra le applicazioni, la segmentazione della clientela in categorie con lo scopo di fornire per ciascuna un migliore servizio o per la promozione mirata di nuovi prodotti, oppure in ambito medico per la valutazione di casi clinici sulla base della sintomatologia e l'analisi epidemiologica.

2.3.4 Analisi delle serie temporali

Questo metodo consiste nell'analizzare un insieme di dati che hanno un andamento variabile nel tempo; lo scopo di questa analisi è trovare delle regolarità nei dati. In particolare, date delle sequenze di dati complessi che hanno un andamento variabile nel tempo, l'obiettivo è trovare sottosequenze che si ripetono frequentemente. Un campo di applicazione è quello economico: è possibile utilizzarlo per studiare l'andamento dei titoli in borsa per individuare, ad esempio, intervalli di tempo in cui un titolo ha un andamento simile.

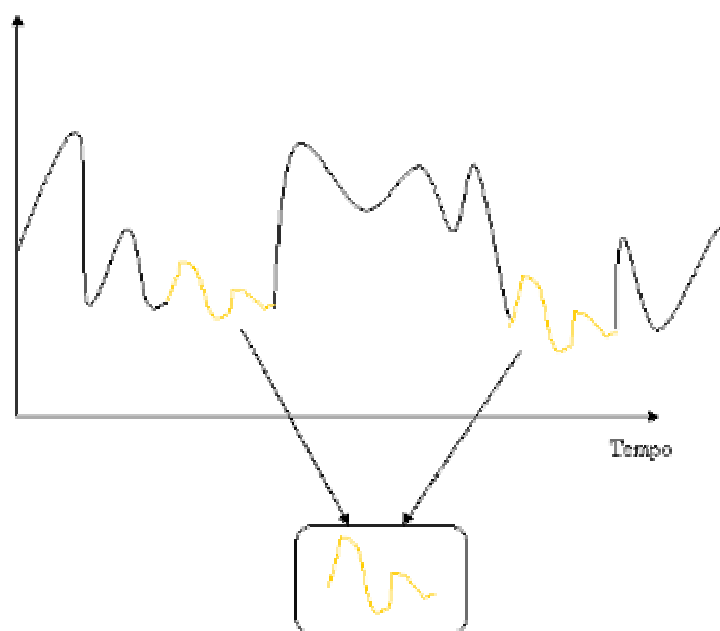


Figura 2.3 Ricerca di serie temporali ricorrenti

2.3.5 Analisi degli Outliers

Un database può contenere data objects che non sono conformi con il comportamento generale o modello dei dati. Questi data objects sono detti *outliers*. Molti metodi di data mining scartano outliers come rumore o eccezioni. Comunque in molte applicazioni come nel rilevamento delle frodi, l'evento raro può essere più interessante dei dati che hanno una certa regolarità. L'analisi degli outliers è riferita come *outlier mining*. Gli outliers possono essere individuati con l'uso di test statistici che assumono un modello di probabilità o distribuzione per i dati, o usando misure di distanza dove gli oggetti che sono sostanzialmente distanti da ogni clusters sono considerati outliers. Piuttosto che usare misure di distanza o statistiche, i metodi basati sulla deviazione identificano outliers esaminando differenze nelle caratteristiche principali degli oggetti di un gruppo.

2.4 Misure di interesse

L'analisi di un insieme di dati con l'ausilio di tecniche di data mining può portare alla scoperta di diverse regole o patterns. In generale il data mining offre buone prestazioni, ma non è detto che tutte le regole estratte siano interessanti. Un pattern è interessante se è facilmente comprensibile dall'utente e se valida un'ipotesi di cui si cerca conferma, con un certo grado di certezza.

Per valutare se una regola sia interessante o meno, si utilizzano delle misure che associano ai patterns un valore che ne indica il grado di interesse. Queste misure, basate sulla struttura dei patterns scoperti e sulla statistica, possono essere di tipo oggettivo o soggettivo.

Tra le misure *oggettive* particolarmente interessanti sono il supporto e la confidenza, già descritte brevemente nel par. 2.3.1. Il supporto rappresenta la percentuale di tuple che soddisfano una regola della forma $X \Rightarrow Y$, cioè la probabilità $P(XUY)$. La confidenza, fornisce il grado di certezza dell'associazione individuata; è definita come la probabilità condizionata che il patterns Y sia vero sapendo che X sia vero, cioè corrisponde alla probabilità $P(Y | X)$.

Nel caso in cui, per valutare la validità di un patterns si ricorra a misure oggettive, è necessario definire una soglia di confidenza: se la misura supera la soglia allora si può concludere che la regola è interessante. La soglia di confidenza è definita dall'utente e dipende dall'applicazione specifica dei dati. Un valore tipico può essere 50%.

Le misure *soggettive* sono basate sulle credenze dell'utente riguardo ai dati: vengono utilizzate per confermare o negare un'ipotesi formulata dall'utente.

In questo caso le misure trovano patterns interessanti se sono inaspettate (contraddicono le credenze dell'utente) o se offrono informazioni strategiche in base alle quali l'utente può agire. Nell'ultimo caso, i patterns sono riferiti come *actionable*.

In generale gli algoritmi di data mining non sono completi, cioè non generano tutti i patterns interessanti. Tuttavia i vincoli dell'utente e le misure di interesse dovrebbero essere usate per circoscrivere la ricerca e questo spesso è sufficiente per assicurare la completezza dell'algoritmo.

2.5 Data Mining, OLAP, statistica e intelligenza artificiale

In alternativa o in modo complementare al data mining, possono essere utilizzate altre procedure di analisi; tra queste le procedure OLAP (On Line Analytical Processing). Si tratta di una tecnica che a differenza del data mining ha alla base un'analisi essenzialmente deduttiva. In tali procedure, l'utente "interroga" il database ponendo una serie di "queries" che vanno a validare o meno un'ipotesi precedentemente formulata. Tuttavia, quando il numero delle variabili cresce, l'utilizzo delle metodologie OLAP diventa sempre più difficoltoso, perché diventa difficile, e anche dispendioso in termini di tempo, formulare delle buone ipotesi da verificare. Il data mining, diversamente da una procedura OLAP, compie un'analisi induttiva: non crea dei modelli ipotetici da validare o confutare ma cerca nuovi modelli. Data mining e OLAP sono strumenti complementari: dato un certo insieme di dati, l'analista può utilizzare tecniche di data mining per scoprire patterns interessanti e sulla base del risultato, può formulare delle interrogazioni per capire meglio i modelli estratti e per raffinare la conoscenza acquisita. Il data mining è legato anche ad altre discipline, quali la statistica e l'intelligenza artificiale. Entrambe, offrono degli strumenti per interpretare in maniera corretta i modelli estratti con il data mining. Quindi tipicamente la risoluzione di problemi che richiedano operazioni di data mining, richiede il supporto di altre tecniche proprie della statistica e dell'intelligenza artificiale.

2.6 Data Mining: applicazioni

Le tecniche di data mining sono applicabili teoricamente a qualsiasi ambito; in generale trovano applicazione tutte le volte che bisogna trattare una grossa mole di dati e si ha l'esigenza di conoscerne il contenuto o scoprire al suo interno informazioni che non si conoscono a priori.

Tra gli ambiti applicativi, particolarmente interessante è quello delle aziende produttrici di beni o servizi. Esse infatti dispongono di grosse quantità di dati, riguardanti ad esempio gli acquisti dei clienti, che se utilizzati in maniera opportuna, possono risultare molto utili, soprattutto ai fini di una crescita economica dell'azienda stessa. Ogni azienda subisce una concorrenza pressante e per sopravvivere, deve essere abile nell'individuare nuovi segmenti di mercato, nello scoprire preferenze e comportamenti dei clienti, nel ridurre eventuali sprechi nella produzione. Per realizzare tutto ciò deve sfruttare adeguatamente i dati in proprio possesso. Con l'ausilio di tecniche di data mining è possibile, ad esempio, studiare le abitudini di acquisto dei clienti individuando i prodotti acquistati congiuntamente (market basket analysis), o individuando raggruppamenti di clienti omogenei in termini di abitudini d'acquisto o caratteristiche socio-demografiche. Sulla base dei risultati, l'azienda può pianificare campagne promozionali che mirino a influenzare i clienti in una certa direzione.

Un altro campo applicativo è rappresentato da quello bancario. Sulla base di un'analisi dei dati, in quest'ambito è possibile individuare per esempio gruppi di clienti a rischio abbandono; in questo caso la banca potrebbe compiere un'attività promozionale verso questa categoria, per esempio attraverso mutui a tasso agevolato.

Il data mining trova applicazione anche in campo assicurativo: attraverso un'analisi dei dati è possibile scoprire comportamenti fraudolenti da parte di alcuni soggetti. Un'analisi in quest'ambito, si basa sul fatto che i dati di un

certo tipo sono caratterizzati da una certa regolarità; la presenza di dati non conformi all'insieme possono mettere in luce una frode.

Un altro campo applicativo è quello delle telecomunicazioni; in questo caso lo scopo è quello di analizzare un insieme di segnali o individuare guasti in una rete di telecomunicazioni, o rilevare anomalie in un sistema di monitoraggio.

Particolarmente interessante è l'applicazione del data mining all'analisi di testi e documenti. Si parla in questo caso di analisi testuale o *text mining*. Si tratta di una particolare applicazione che consente di individuare sequenze di parole (patterns), che accomunano o caratterizzano un insieme di documenti e che consentono perciò il raggruppamento tematico. Risulta particolarmente utile quando si deve analizzare il contenuto di una collezione di documenti (anche provenienti da fonti eterogenee). L'individuazione di gruppi tematici, consente di dare un'organizzazione all'informazione disponibile e di individuare argomenti secondari, che anche ad una lettura attenta potrebbero sfuggire. Le relazioni, inoltre, mettono in evidenza legami tra argomenti apparentemente eterogenei ma che hanno una terminologia comune.

I campi applicativi del data mining sono numerosi e attualmente è in fase di studio l'applicabilità in settori nuovi. Uno studio di questo tipo è lungo e spesso i dati, su cui si cerca di applicare il data mining, sono in un formato complesso e diverso da quello relazionale. Questo richiede un adattamento delle tecniche di estrazione della conoscenza a questi nuovi dati ed in alcuni casi, quando una tale applicazione non è possibile, lo sviluppo di nuovi metodi di analisi. In un tale contesto si inserisce il *data mining spazio-temporale* per il trattamento di entità che si muovono nello spazio. Rientra in questo ambito lo studio delle traiettorie compiute in un ambiente. Può essere interessante, ad esempio, individuare gruppi di traiettorie simili. Uno studio di questo tipo richiede lo sviluppo di algoritmi di clustering specifici per il trattamento di entità che si muovono in uno spazio. Altre tecniche adottate per il trattamento di dati più

complessi rispetto a quelli relazionali sono rappresentate dal *Graph Mining* e dal *XML Mining*.

3. Clustering e classificazione

Il data mining ha diversi campi di applicazione che si estendono dalla biologia alle telecomunicazioni, dall'economia al campo finanziario, dall'astronomia al web. Lo scopo di questo lavoro di tesi è sperimentarne l'applicabilità ad un campo nuovo: la realtà virtuale.

Prima di cimentarsi nella progettazione e nello sviluppo di metodi per trattare i dati di realtà virtuale, si verifica se, utilizzando le tecniche di data mining tradizionali su informazioni "virtuali", si ottengono risultati soddisfacenti. A tale scopo si è scelto di applicare il clustering e la classificazione ai dati estratti durante un particolare esperimento di realtà virtuale (descritto nel capitolo 5). Lo scopo in questo capitolo è descrivere in maniera più dettagliata clustering e classificazione rispetto a quanto fatto in precedenza, nel capitolo 2.

3.1 Clustering: Che cos'è?

Come abbiamo già brevemente discusso nel capitolo 2, il clustering è una tecnica di data mining che dato un insieme di oggetti di un certo tipo li raggruppa in classi o *clusters*. La costruzione di questi raggruppamenti è fatta in modo che oggetti appartenenti allo stesso cluster siano molto simili tra loro e siano diversi da qualunque oggetto appartenente ad altri clusters. L'obiettivo del clustering è quindi raggruppare un insieme di oggetti massimizzando la similarità *intra-cluster* e minimizzando la similarità *inter-cluster*. Il processo di clustering può essere suddiviso convenzionalmente in due fasi:

1. Individuazione dei clusters o gruppi.

2. Descrizione di ciascun cluster sulla base dei valori degli attributi che caratterizzano gli oggetti che vi appartengono.

Un buon metodo di clustering deve godere di certe proprietà:

- *Scalabilità.* Deve offrire buone prestazioni indipendentemente dal numero di oggetti da raggruppare.
- *Capacità di trattare differenti tipi di dati.* Deve essere capace di elaborare dati di qualsiasi tipo, non solo dati numerici ma anche dati binari, categorici, ordinali e di tipo misto.
- *Capacità di scoprire clusters con forma arbitraria.* Questa proprietà dipende dal tipo di funzioni utilizzate nell'algoritmo di clustering per determinare la similarità tra oggetti. Ad esempio gli algoritmi che utilizzano la distanza di Manhattan o la distanza euclidea tendono ad individuare clusters sferici.
- *Richiesta minima di valori in input.* Molti algoritmi di clustering richiedono all'utente di fornire alcuni parametri di input (come per esempio il numero di clusters da individuare). Il numero di parametri richiesti dall'algoritmo deve essere il minimo possibile cosicché i risultati non siano influenzati da input errati.
- *Capacità di trattare "rumore" e valori mancanti.* I dati che descrivono gli oggetti da raggruppare, possono contenere informazioni che costituiscono rumore, che non danno cioè alcun informazione riguardo l'oggetto; l'algoritmo di clustering dovrebbe essere in grado di individuare il rumore e scartarlo. Un buon algoritmo dovrebbe avere anche un modo per trattare gli attributi i cui valori sono stati omessi; una soluzione a questo problema è assegnargli un valore ipotetico o quello che occorre più frequentemente per l'attributo.
- *Insensibilità all'ordine in cui si forniscono i dati.* Il risultato del processo di clustering dovrebbe essere indipendente dall'ordine nel quale si forniscono i dati in input.

- *Alta dimensionalità.* Un buon algoritmo di clustering dovrebbe essere efficiente indipendentemente dal numero di attributi di ciascun oggetto.
- *Interpretabilità e usabilità.* I risultati del processo di clustering devono essere facilmente interpretabili e usabili.

3.2 Clustering: rappresentazione dei dati

Il processo di clustering richiede che i dati relativi agli oggetti da raggruppare siano organizzati in un certo modo. Esistono due metodi per organizzare i dati: la matrice dei dati (*data matrix*) e la matrice di dissimilarità (*dissimilarity matrix*). Il primo metodo richiede di rappresentare i dati mediante una tabella relazionale caratterizzata da tante righe quanti sono gli oggetti e tante colonne quanti sono gli attributi.

Nell'ipotesi di dover clusterizzare n oggetti ciascuno avente p attributi, si utilizza una tabella relazionale $n \times p$.

La matrice di dati avrà la forma seguente:

$$\begin{array}{cccc}
 x_{11} & x_{12} & \dots & x_{1p} \\
 x_{21} & \dots & \dots & x_{2p} \\
 \dots & \dots & \dots & \dots \\
 x_{n1} & \dots & \dots & x_{np}
 \end{array} \tag{3.1}$$

Il secondo metodo prevede, dati n oggetti, l'utilizzo di una tabella $n \times n$ in cui l'elemento $d(i,j)$ è una misura della dissimilarità tra gli oggetti i e j . La matrice di dissimilarità sarà di questo tipo:

$$\begin{array}{cccc}
 0 & d(1,2) & \dots & d(1,n) \\
 d(2,1) & 0 & \dots & d(2,n) \\
 \dots & \dots & \dots & \dots \\
 d(n,1) & d(n,2) & \dots & 0
 \end{array} \tag{3.2}$$

Tanto più piccolo è il valore $d(i,j)$, tanto minore è la distanza tra gli oggetti i e j . Ovviamente valgono alcune proprietà banali, vale a dire $d(i,j) = d(j,i)$ e $d(i,i) = 0$.

Nel caso in cui gli oggetti sono rappresentati con una matrice di dati, prima di realizzare il clustering, può essere utile ricavarsi anche la matrice di dissimilarità.

3.3 Clustering: dissimilarità

La costruzione della matrice di dissimilarità per un insieme di oggetti, richiede la scelta di un metodo per calcolare la distanza tra gli oggetti stessi. La scelta della misura di similarità è di primaria importanza in quanto influisce insieme allo specifico algoritmo di clustering utilizzato sulla qualità del risultato. La scelta del metodo dipende dal tipo dei dati.

3.3.1 Variabili numeriche

Una variabile numerica è una variabile che assume valori continui su una scala lineare. Tipici esempi di variabili numeriche sono il peso e l'altezza.

La dissimilarità tra oggetti descritti da variabili numeriche viene calcolata come distanza.

Tra le misure di distanza, particolarmente interessante è la *distanza Euclidea* definita come

$$d(i, j) = \sqrt{\left(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \dots + |x_{ip} - x_{jp}|^2 \right)}, \quad (3.3)$$

dove $i=(x_{i1}, \dots, x_{ip})$ e $j=(x_{j1}, \dots, x_{jp})$ sono due oggetti p-dimensionali cioè descritti da p attributi.

Un'altra misura di distanza è quella di Manhattan definita come:

$$d(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{ip} - x_{jp}|. \quad (3.4)$$

Un'altra misura è la *distanza di Minkowski*: si tratta di una generalizzazione della distanza euclidea e della distanza di Manhattan. E' definita come:

$$d(i, j) = \left(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \dots + |x_{ip} - x_{jp}|^q \right)^{1/q}. \quad (3.5)$$

Rappresenta la distanza di Manhattan quando $q=1$, la distanza euclidea quando $q=2$.

Tutti questi metodi per calcolare la distanza tra oggetti, soddisfano alcune proprietà matematiche fondamentali:

- $d(i,j) \geq 0$: la distanza tra due oggetti è un valore non negativo.
- $d(i,i) = 0$: la distanza di un oggetto da se stesso è zero.
- $d(i,j) = d(j,i)$: la distanza è una funzione simmetrica.
- $d(i,j) \leq d(i,h) + d(h,j)$: nota come *disuguaglianza triangolare*; per questa proprietà la distanza tra due oggetti i e j non è più grande della distanza tra i e j passando per un terzo oggetto h .

Poiché ciascun oggetto generalmente è descritto da più attributi, si può dare un peso diverso a ciascuno di essi, influenzando il calcolo della distanza in un certo modo. Per esempio nel caso in cui il clustering riguardi i dati di giocatori di basket, si può decidere di dare maggiore peso alla variabile altezza.

In questo caso, se come distanza si utilizza quella euclidea, la formula sarà la seguente

$$d(i, j) = \sqrt{w_1|x_{i1} - x_{j1}|^2 + w_2|x_{i2} - x_{j2}|^2 + \dots + w_p|x_{ip} - x_{jp}|^2}, \quad (3.6)$$

dove w_k , $1 \leq k \leq p$ è il peso associato all'attributo k .

Nel caso in cui si vuole dare uguale peso a tutti gli attributi, può essere necessario standardizzare i dati in modo che non siano influenzati dalle unità di misura in cui sono espressi i valori dei vari attributi. La rappresentazione degli attributi in unità di misura differenti può condurre a misure di distanza differenti. Dati i valori per una variabile f , la standardizzazione può essere fatta in questo modo:

1. Si calcola la deviazione media assoluta s_f , come

$$s_f = \frac{1}{n} * (|x_{1f} - m_f| + |x_{2f} - m_f| + \dots + |x_{nf} - m_f|), \quad (3.7)$$

dove x_{1f}, \dots, x_{nf} sono n valori di f , ed m_f è la media dei valori di f cioè

$$m_f = \frac{1}{n} * (x_{1f} + \dots + x_{nf}). \quad (3.8)$$

2. Ogni x_{if} viene sostituito con il valore standardizzato o *z-score*,

$$z_{if} = \frac{x_{if} - m_f}{s_f}. \quad (3.9)$$

3. Si calcola la distanza tra gli oggetti applicando una tra le distanze descritte (euclidea, Minkowski o Manhattan).

3.3.2 Variabili binarie

Una variabile binaria ha due valori possibili, 0 o 1; quando vale 0 significa che la variabile è assente, quando vale 1 che è presente. Per esempio, nell'ipotesi di raggruppare i dati relativi ai pazienti di un ospedale, ciascuno di essi può aver associato un attributo fumatore, di tipo binario, che vale 1 se il paziente fuma e vale 0 altrimenti.

Una variabile binaria può essere simmetrica o asimmetrica. Una variabile è simmetrica se i due stati hanno lo stesso peso. Per esempio la variabile sesso, che ha due valori possibili, maschio o femmina, è una variabile binaria simmetrica.

Una variabile binaria è asimmetrica se i due stati non hanno lo stesso peso; un esempio è rappresentato dal risultato di un test per verificare la presenza di una malattia; in questo caso ha più importanza lo stato 1 rispetto allo stato 0.

Nel caso in cui si debba calcolare la dissimilarità tra una coppia di oggetti descritti da variabili binarie, si utilizza una tabella di contingenza 2 x 2 che ha una forma del tipo riportato in figura 3.1.

		Oggetto j		
		1	0	sum
Oggetto i	1	q	r	$q + r$
	0	s	t	$s + t$
		$q + s$	$r + t$	p

Figura 3.1 Tabella di contingenza per variabili binarie

Le variabili riportate in tale tabella vengono interpretate in questo modo:

- q è il numero di variabili che sono uguali a 1 per entrambi gli oggetti i e j ;
- r è il numero di variabili che sono uguali a 1 per l'oggetto i e uguali a 0 per l'oggetto j ;
- s è il numero di variabili che sono uguali a 0 per l'oggetto i e uguali a 1 per l'oggetto j ;
- t è il numero di variabili che sono uguali a 0 per entrambi gli oggetti i e j ;
- p è il numero totale di variabili cioè $p = q + r + s + t$.

Nel caso di variabili simmetriche si parla di *similarità invariante*; per calcolare la dissimilarità tra due oggetti i e j descritte da variabili di questo tipo si utilizza un coefficiente di matching definito come

$$d(i, j) = (r + s) / (q + r + s + t). \quad (3.10)$$

Nel caso di variabili asimmetriche si parla di *similarità variante*. Per questo tipo di similarità il coefficiente più conosciuto è quello di *Jaccard*, in cui il numero di match negativi t non è considerato importante e quindi è ignorato nella computazione della dissimilarità. In questo caso la dissimilarità tra l'oggetto i e l'oggetto j è calcolata come

$$d(i, j) = (r + s) / (q + r + s). \quad (3.11)$$

3.3.3 Variabili nominali

Una variabile nominale è una variabile che può prendere un numero finito di valori. E' una generalizzazione delle variabili binarie. La dissimilarità tra due oggetti i e j , descritti da variabili nominali è calcolata come

$$d(i, j) = (p - m) / p, \quad (3.12)$$

dove p è il numero totale di variabili, mentre m è il numero di match (cioè il numero di variabili per le quali i e j hanno lo stesso valore).

Una variabile nominale può essere decodificata con un gran numero di variabili binarie; a tale scopo si introduce una nuova variabile binaria per ciascuno dei valori assunti dalla variabile nominale. Una trasformazione di questo tipo consente poi di utilizzare le stesse misure già descritte per le variabili binarie.

3.3.4 Variabili ordinali

Una variabile ordinale è una variabile nominale sui cui valori è definito un ordinamento. Se tra gli attributi di un insieme di oggetti da clusterizzare, ci sono variabili di questo tipo, occorre trasformare questi attributi in variabili numeriche in modo poi da trattarle come tali ai fini del calcolo della similarità tra gli oggetti. Per realizzare questa trasformazione, sono previsti diversi passi:

- Supponiamo che x_{if} sia il valore dell'attributo f -esimo di tipo ordinale per l'oggetto i , che può assumere M_f valori diversi. Sotto queste ipotesi, per ogni oggetto i , ciascun x_{if} viene rimpiazzato con $r_{if} \in \{1, \dots, M_f\}$.
- Dal momento che un oggetto può essere descritto attraverso più variabili ordinali e poiché ciascuna di esse può avere un numero diverso di valori, si mappano i valori di ogni variabile nell'intervallo $[0,1]$. In particolare, si mappa l' i -esimo valore della f -esima variabile con

$$z_{if} = \frac{r_{if} - 1}{M_f - 1} \quad (3.13)$$

- Eseguita questa trasformazione, si può calcolare la dissimilarità tra variabili ordinali trattandole come variabili numeriche, utilizzando per l'attributo ordinale f dell'oggetto i , z_{if} anziché x_{if} .

3.3.5 Variabili di tipo misto

Il processo di clustering viene applicato in questo caso ad un insieme di oggetti p -dimensionali. Nella maggior parte dei casi è difficile che gli attributi abbiano tutti lo stesso tipo. Esistono differenti approcci per trattare oggetti descritti da variabili di tipo diverso. Uno di questi prevede di raggruppare le variabili per tipo ed eseguire il calcolo della dissimilarità in maniera diversa per ogni gruppo di variabili. Un approccio di questo tipo in generale non offre buoni risultati perché è improbabile che analisi separate per gruppi di variabili siano compatibili.

Un approccio alternativo consiste nel processare tutti i tipi di variabili insieme; in questo caso si utilizza una formula pesata per rappresentare gli effetti delle distanze calcolate con metodi diversi.

In questo caso la dissimilarità tra l'oggetto i e l'oggetto j è definita come

$$d(i, j) = \frac{\sum_{f=1}^p \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^p \delta_{ij}^{(f)}} \quad (3.14)$$

dove

- Se f è binaria asimmetrica e x_{if} o x_{jf} non esistono nei due oggetti i e j oppure $x_{if} = x_{jf} = 0$ allora $\delta_{ij}^{(f)} = 0$; altrimenti $\delta_{ij}^{(f)} = 1$.
- Nel caso in cui f sia binaria o nominale e $x_{if} = x_{jf}$ allora $d_{ij}^{(f)} = 0$; altrimenti $d_{ij}^{(f)} = 1$.
- Se f è ordinale si calcolano i ranghi r_{if} e si tratta z_{if} (calcolata come in 3.13) come numerica.

Rispettando queste regole di trasformazione, si mappano i valori di tutte le variabili nello stesso intervallo $[0,1]$.

3.4 Classificazione dei principali metodi di clustering

In letteratura esistono diversi algoritmi di clustering; la scelta dell'algoritmo da utilizzare dipende dall'applicazione specifica e dal tipo di dati.

Gli algoritmi di clustering si suddividono in 5 gruppi:

- *Metodi di partizionamento.* Dato un database di n oggetti, un metodo di partizionamento costruisce k partizioni dei dati, ciascuna delle quali rappresenta un cluster. Si suddividono i dati in k gruppi che devono soddisfare alcuni requisiti:

- 1) Ogni gruppo deve contenere almeno un elemento.
- 2) Un elemento deve appartenere esattamente ad un gruppo.

Dato un valore k , un tipico algoritmo di partizionamento crea inizialmente un'unica partizione. Poi in maniera iterativa gli oggetti vengono spostati in gruppi diversi. Il concetto alla base del partizionamento è che oggetti appartenenti allo stesso cluster siano vicini o simili ad ogni altro, mentre oggetti appartenenti a clusters diversi siano molto lontani o diversi.

- *Metodi gerarchici.* Un metodo gerarchico crea una decomposizione gerarchica di un insieme di oggetti. Un metodo gerarchico può essere *agglomerativo* o *divisivo* a seconda di come viene fatta la decomposizione gerarchica. Un approccio agglomerativo o *bottom-up*, inizialmente considera ogni oggetto indipendentemente dagli altri; ogni elemento rappresenta un gruppo di cui è l'unico membro. Ad ogni passo uno o più oggetti vengono raggruppati insieme, si agglomerano più gruppi fino ad arrivare al punto in cui si ha un unico gruppo o si è verificata una condizione di terminazione. Un approccio divisivo o *top-down*, inizia con tutti gli oggetti nello stesso cluster. Ad ogni iterazione successiva, un cluster è suddiviso in clusters più piccoli, finché eventualmente ogni oggetto costituisce un gruppo a se stante o finché non sono soddisfatte delle condizioni di terminazione.

- *Metodi density-based.* Questi metodi consentono di individuare clusters di forma arbitraria, diversamente da altri metodi; per esempio i metodi di partizionamento che utilizzano funzioni di distanza per valutare la dissimilarità, individuano clusters di forma sferica. I metodi density-based si basano sul concetto di *densità*. Con l'utilizzo di metodi density-based, si considera l'intorno di un oggetto: tutti gli oggetti entro un certo raggio vanno a costituire un cluster. Un cluster cresce finché la densità, ossia il numero di oggetti che vi appartengono, non raggiunge una certa soglia. Nello spazio in cui sono disposti gli oggetti, le zone caratterizzate da alta densità di punti (tanti oggetti) rappresentano clusters; le zone caratterizzate da bassa densità separano i clusters e rappresentano rumore.
- *Metodi grid-based.* Questi metodi definiscono una griglia su tutto lo spazio in cui sono localizzati gli oggetti. Tutte le operazioni di clustering sono eseguite sulla struttura griglia. Il principale vantaggio di questo approccio è la velocità del tempo di esecuzione che tipicamente è indipendente dal numero di oggetti e dipende solo dal numero di celle che definiscono la griglia.
- *Metodi basati su modello.* I metodi basati su modello prevedono la definizione di un modello per ciascun cluster; ciascun oggetto viene assegnato al cluster di cui soddisfa il modello.

Nei prossimi paragrafi approfondiremo i metodi di partizionamento e quello gerarchico che sono stati impiegati in questa tesi.

3.5 Metodi di partizionamento: K– Means

Dato un database di n oggetti, e un numero di clusters k da formare, un algoritmo di partizionamento organizza gli oggetti in k partizioni ($k \leq n$), dove ogni partizione rappresenta un cluster. I metodi di partizionamento più conosciuti sono il k-means e il k-medoid.

L'algoritmo k-means prende come parametro di input k e partiziona un insieme di n oggetti in k clusters in modo tale che la similarità intra-cluster sia alta, mentre la similarità inter-cluster sia bassa.

L'algoritmo k-means lavora in questo modo.

- 1) Per prima cosa si selezionano k oggetti in maniera random, ciascuno dei quali inizialmente rappresenta il centro dei k clusters. Ciascuno degli oggetti rimanenti viene assegnato al cluster al quale risultano più simili basandosi sulla distanza tra ciascun oggetto e il valor medio di ciascun cluster. Il valore medio di un cluster viene calcolato come media delle posizioni degli oggetti che vi appartengono. Inizialmente, poiché ciascun cluster ha un solo elemento il valor medio corrisponderà alla posizione dell'unico elemento che ne fa parte.
- 2) Dopo questa prima clusterizzazione si determinano i nuovi valori medi per ogni cluster.

Questo processo viene eseguito in maniera iterativa fino a quando non si verifica che la configurazione dei clusters non cambia o cambia poco.

In figura 3.2 è riportato lo pseudocodice di un tipico algoritmo di partizionamento di tipo k-means.

Algorithm : *K-means*. The k-means algorithm for partitioning based on the mean value of the objects in the cluster.

Input: The number of clusters k , and a database containing n objects.

Output: A set of k clusters which minimizes the squared-error criterion

Method:

1. arbitrarily choose k objects as the initial cluster centers;
2. Repeat
3. (re)assign each object to the cluster to which the object is the most similar,
4. based on the mean value of the objects in the cluster,
5. update the cluster means, i.e., calculate the mean value of the objects for each cluster;
6. Until no change;

Figura 3.2 Pseudocodice dell'algoritmo k-means

Per valutare il clustering prodotto dall'algoritmo, si utilizza una funzione di errore quadratico definita come

$$E = \sum_{i=1}^k \sum_{p \in C_i} |p - m_i|^2, \quad (3.15)$$

dove E è la somma dell'errore quadratico per tutti gli oggetti nel database, p è il punto nello spazio che rappresenta l'oggetto dato ed m_i è la media del cluster C_i . Dati due clustering ottenuti con diverse esecuzioni del k-means, è migliore quello che minimizza l'errore quadratico. L'algoritmo offre buoni risultati quando i clusters risultano abbastanza compatti e ben separati dagli altri. K-means risulta abbastanza efficiente; ha una complessità dell'ordine di $O(knt)$, dove n è il numero totale di oggetti, k è il numero di clusters e t è il numero di iterazioni. Normalmente vale $k \ll n$ e $t \ll n$.

Il metodo spesso termina ad un ottimo locale; un ottimo globale può essere individuato con altre tecniche.

Pur essendo sufficientemente efficiente in termini di complessità, il k-means ha anche dei punti deboli:

- Può essere applicato solo quando il tipo di dato permette di definire la media (che serve per determinare i centroidi del cluster); si hanno quindi dei problemi per esempio con i dati categorici.

- Bisogna specificare in anticipo il numero di clusters k .
- Ha problemi quando i clusters hanno differenti dimensioni e densità e quando i dati contengono outliers.

Esistono diverse varianti del metodo k -means che differiscono nella selezione del k iniziale, nel calcolo della dissimilarità o nella strategia per calcolare la media del cluster.

Esempio 3.1

In figura 3.3 è riportato un esempio di applicazione del k -means ad un insieme di oggetti distribuiti in uno spazio. Inizialmente si scelgono k oggetti in maniera random che rappresentano i centroidi dei k clusters. Gli altri oggetti vengono raggruppati insieme al centroide a cui risultano più simili (vicini). Una volta assegnati tutti gli oggetti si ricalcola il centro di ciascun cluster e si ridistribuiscono gli oggetti tra i clusters.

Il centroide di ciascun clusters è indicato con “+”.

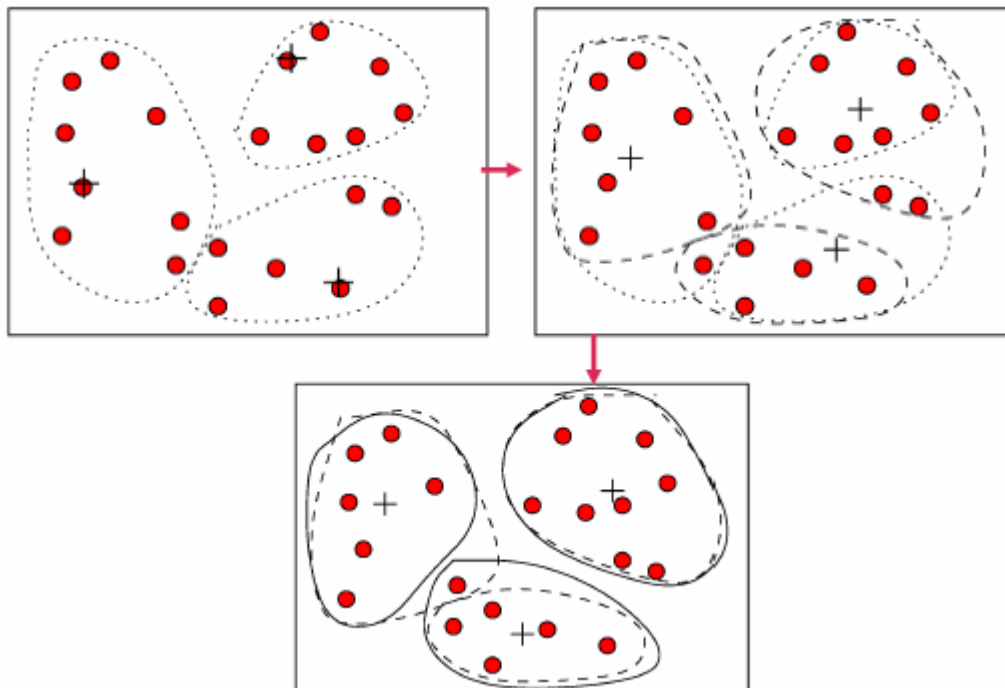


Figura 3.3 Esempio di clustering realizzato con il k -means

3.6 Metodo gerarchico

Un algoritmo di clustering gerarchico lavora raggruppando gli oggetti in un albero di clusters. I metodi di clustering gerarchico possono essere classificati in metodi di tipo *agglomerativo* e metodi di tipo *divisivo*, a seconda che il clustering venga fatto attraverso un partizionamento ricorsivo (*approccio top-down*) o una combinazione di clusters esistenti (*approccio bottom-up*).

Un metodo di clustering agglomerativo utilizza un approccio bottom-up, piazzando ogni oggetto nel proprio cluster e raggruppando questi clusters in clusters sempre più grandi, finché tutti gli oggetti non sono in un unico cluster o finché non sono soddisfatte certe condizioni di terminazione.

I metodi di clustering gerarchico divisivo utilizzano una strategia top-down, con la procedura inversa a quella di un metodo di clustering agglomerativo e iniziando con tutti gli oggetti in un unico cluster. Suddividono poi i clusters in sotto-clusters sempre più piccoli, finché ogni oggetto non forma un cluster a se stante o finché non sono soddisfatte certe condizioni di terminazione.

Il processo di merging si ripete finché tutti gli oggetti vengono eventualmente uniti per formare un unico cluster. Sia nel clustering agglomerativo che in quello divisivo, l'utente può specificare il numero di clusters desiderato come condizione di terminazione.

Si considerano quattro misure per la distanza tra clusters, dove m_i è la media per il cluster c_i , n_i è il numero di oggetti in c_i e $|p - p'|$ è la distanza tra due oggetti o punti p e p' .

- *Distanza minima:* $d_{\min}(c_i, c_j) = \min_{p \in C_i, p' \in C_j} |p - p'|$ (3.16)

- *Distanza massima:* $d_{\max}(c_i, c_j) = \max_{p \in C_i, p' \in C_j} |p - p'|$ (3.17)

- *Distanza media:* $d_{\text{mean}}(c_i, c_j) = |m_i - m_j|$ (3.18)

- *Average distance:* $d_{\text{avg}}(c_i, c_j) = \frac{1}{n_i * n_j} \sum_{p \in C_i} \sum_{p' \in C_j} |p - p'|$ (3.19)

Nel caso in cui per valutare la similarità si utilizzi la distanza minima, si parla di approccio *nearest-neighbor*; i clusters individuati hanno una forma allungata e sottile.

Nel caso in cui si utilizzi la distanza massima, si parla di approccio *farthest-neighbor*; i clusters individuati hanno forma sferica.

Il metodo di clustering gerarchico incontra difficoltà nella selezione di punti di merge o di split. Una decisione di questo tipo è critica perché una volta che un gruppo viene unito ad un altro o viene diviso, il processo al passo successivo lavora su questi nuovi clusters. Nel caso in cui ci si renda conto che l'operazione effettuata non è corretta non è possibile disfare quanto fatto, scambiando gli oggetti tra i clusters. In generale offre prestazioni migliori rispetto ad un metodo di partizionamento nel caso venga applicato su pochi dati.

Esempio 3.2

In figura 3.4 è riportato un esempio di clustering realizzato con un algoritmo gerarchico di tipo agglomerativo. L'insieme di oggetti è rappresentato da $\{1,2,3,4,5,6\}$. Inizialmente ciascun oggetto costituisce un cluster di cui è l'unico membro. Ad ogni iterazione successiva si uniscono più clusters in base ad una funzione di distanza. La procedura termina quando nel processo di unificazione si arriva ad ottenere un unico cluster oppure è verificata un'altra condizione di terminazione.

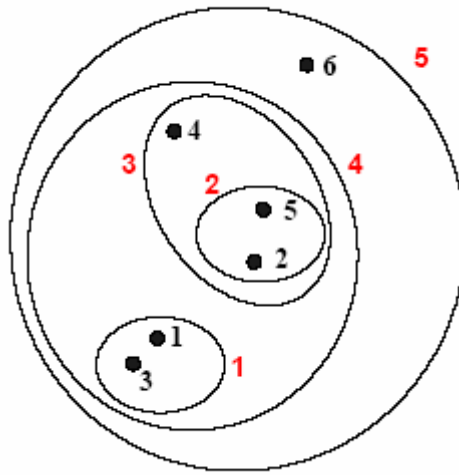


Figura 3.4 Esempio di clustering realizzato con algoritmo gerarchico

3.7 Clustering di traiettorie

Negli ultimi anni alcuni studi di data mining sono stati condotti per l'estrazione di conoscenza da oggetti in movimento. Un'entità che si muove nello spazio compie una traiettoria modellata da una sequenza di coordinate spaziali che, nel caso di spazio bidimensionale, viene rappresentata come:

$$((x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)) \quad (3.20)$$

dove la coppia (x_i, y_i) rappresenta la posizione nello spazio dell'oggetto all'istante i . Un problema che è ancora tema di studio per gli esperti di data mining è quello di determinare gruppi di entità che compiono la stessa traiettoria ovvero della realizzazione del clustering di traiettorie. In questo caso due o più traiettorie vengono raggruppate insieme se sono simili. Il clustering di entità in movimento è più complesso rispetto a quello realizzato su oggetti statici e pone una serie di problemi nella definizione del concetto di similarità. Nel caso di oggetti statici la similarità viene valutata in termini di distanza. Lo stesso criterio può essere utilizzato nel clustering di traiettorie, raggruppando insieme le traiettorie che risultano vicine sia in termini spaziali che temporali. La similarità in questo contesto può essere intesa anche in termini di movimento simile: questa interpretazione consente di raggruppare insieme traiettorie che hanno un andamento simile in regioni spaziali diverse. Nel valutare la similarità tra due traiettorie si può tener conto di tutti i punti che le definiscono o solo di una parte. Due traiettorie, ad esempio, possono coincidere nei punti iniziali e finali ma internamente possono risultare molto diverse.

3.7.1 Distanza euclidea vs LCSS

Esistono diversi metodi per valutare la similarità tra traiettorie e la scelta su quale utilizzare dipende dallo scopo per il quale si realizza il clustering.

Tra i tanti esistenti, particolarmente interessanti sono LCSS (Longest Common SubSequence) e la distanza euclidea.

Il metodo LCSS [Vlac02, Vlac04] viene utilizzato tipicamente con algoritmi di clustering gerarchici e prevede che le traiettorie vengano confrontate a coppie. Consente di individuare, tra due sequenze di coordinate relative a due traiettorie distinte, la più lunga sottosequenza simile che le accomuna. I principali vantaggi offerti da questo metodo sono:

- La possibilità di confrontare traiettorie di lunghezza diversa in quanto il confronto non viene fatto tra coppie di punti;

- L'insensibilità agli outliers in quanto si considerano le porzioni di traiettorie simili.

Il principale svantaggio di questo metodo è che la nozione di similarità utilizzata non è transitiva. Se A , B , C sono traiettorie tali che A è simile a B e B è simile a C , non è detto che A sia simile a C .

Una rappresentazione di una situazione di questo tipo è riportata in figura 3.5.

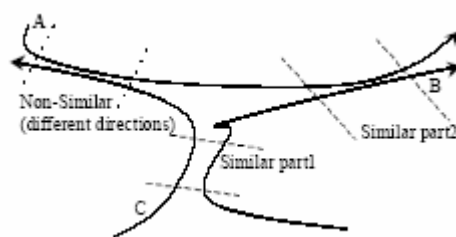


Figura 3.5 Non transitività della nozione di similarità per LCSS

La distanza euclidea richiede l'utilizzo della formula 3.3 che prevede di confrontare i punti di una traiettoria con i corrispondenti punti di un'altra traiettoria. Il principale vantaggio nasce dalla semplicità di calcolo di questa informazione. Il principale svantaggio è l'impossibilità di confrontare due traiettorie di lunghezza diversa. Si può ovviare a questo inconveniente troncando le sequenze di coordinate più lunghe o "allungando" quelle più corte. In quest'ultimo caso si ripetono i punti iniziali e finali delle traiettorie più corte finché non hanno una lunghezza pari a quella massima presente.

L'algoritmo di clustering utilizzato in questo lavoro di tesi prevede l'utilizzo della distanza euclidea. Una descrizione dettagliata di questo algoritmo e della strategia per applicare la distanza euclidea a traiettorie di lunghezza diversa è fornita in [Nan02].

3.8 Classificazione: che cos'è?

La classificazione è un processo che, dato un insieme di oggetti ciascuno dei quali ha certe caratteristiche e appartiene ad una particolare classe, è in grado di definire dei modelli che descrivono tutte le possibili classi. Ogni oggetto può essere rappresentato come un record avente diversi campi; ogni campo conterrà il valore per un particolare attributo. Se tra gli attributi di un oggetto c'è anche l'attributo che indica la sua classe di appartenenza, si dice che l'oggetto ha un'etichetta di classe.

Tutti gli oggetti di uno stesso tipo che hanno un'etichetta di classe, possono essere utilizzati come input per un algoritmo di classificazione: in riferimento a questo insieme di oggetti si utilizza il termine *training set*. Il training set potrebbe avere la forma seguente:

	<i>Attributo(A_1)</i>	<i>Attributo(A_2)</i>	<i>Attributo(A_p)</i>	<i>Classe</i>
<i>Record 1</i>
<i>Record 2</i>
.....
<i>Record n</i>

Figura 3.6 Classificazione: training set

In figura 3.6 sono rappresentati n *records* (oggetti), ciascuno dei quali ha p *attributi* (caratteristiche) con determinati valori; i records che hanno la stessa etichetta di classe appartengono alla stessa classe. Lo scopo della classificazione è quello di trovare delle affinità (in termini dei valori degli attributi) tra i vari oggetti che appartengono ad una stessa classe. L'output dell'algoritmo di classificazione sarà un modello, cioè una descrizione degli oggetti che appartengono ad una stessa classe espressa tramite vincoli sui valori degli attributi. Si avranno tanti modelli quante sono le classi di oggetti.

Questi modelli vengono utilizzati in una fase successiva per predire la classe di appartenenza di un oggetto privo di etichetta di classe; se i valori dei suoi attributi soddisfano i vincoli definiti nel modello di una particolare classe x , allora l'oggetto appartiene a quella classe e gli verrà assegnata come etichetta di classe x .

La classificazione è una tecnica che ha molteplici applicazioni nella vita reale: le banche, per esempio, la utilizzano per individuare le categorie di clienti a cui concedere un prestito (in questo caso ogni record è etichettato con “*buon creditore*” o “*cattivo creditore*” e gli attributi sono i dati dei clienti relativi all'età, al reddito, ecc.).

3.9 Classificazione: fasi preliminari

Esistono diversi modi per realizzare la classificazione. Qualunque sia il metodo utilizzato, è prevista una fase preliminare alla classificazione vera e propria che ha come scopo quello di garantire l'accuratezza dei modelli definiti nella fase successiva. Questa fase preliminare prevede tre operazioni fondamentali:

- *Pulizia dei dati.* Nel training set ci possono essere dati che sono “rumore” e attributi di oggetti per i quali non è stato definito un valore. Per quando riguarda il rumore, l'attività di pulizia dei dati ha come scopo quello di individuare i dati di questo tipo ed eliminarli; possono infatti compromettere l'attendibilità del modello restituito dal classificatore. Il problema degli attributi “privi di valore”, viene risolto assegnando a questi attributi un valore di default o il valore che occorre più spesso (per l'attributo) tra le tuple del training set.
- *Trasformazione dei dati.* Gli attributi che descrivono un particolare oggetto possono avere valori continui o discreti. La classificazione è più accurata se i dati sono discreti. Se tra i dati ci fosse, ad esempio, un attributo reddito a

valori continui, lo si potrebbe generalizzare con un attributo a valori discreti (ad esempio con valori “basso”, “medio”, “alto”).

- *Analisi di rilevanza.* Alcuni attributi che descrivono un oggetto possono essere ridondanti, forniscono quindi informazione già contenuta in altri attributi; altri dati potrebbero essere addirittura irrilevanti, la loro presenza non ha alcun effetto sulla classificazione. Tutti gli attributi che rientrano in queste due categorie possono essere omessi dal processo di classificazione: l'accuratezza del modello non ne viene compromessa.

3.10 Proprietà di un buon algoritmo di classificazione

Un algoritmo di classificazione può essere più o meno efficiente di un altro. L'efficienza viene valutata in termini di:

- *Accuratezza:* la misura di quanto un modello predice correttamente l'etichetta di classe di oggetti che ne sono privi.
- *Robustezza:* capacità del modello di predire correttamente anche in presenza di rumore o attributi privi di valore.
- *Velocità:* il tempo necessario per costruire i modelli e impiegarli per individuare la classe di un oggetto.
- *Scalabilità:* capacità dell'algoritmo di determinare modelli accurati anche con training set di grosse dimensioni.
- *Interpretabilità:* legata alla semplicità e comprensibilità dei modelli costruiti con l'algoritmo.

Tra questi aspetti, quello più importante è l'accuratezza. Questa misura non viene utilizzata solo per verificare quanto correttamente un classificatore predice l'etichetta di classe di un dato record, ma viene utilizzata anche per confrontare algoritmi di classificazione diversi. Se per un certo tipo di oggetto la classificazione può essere realizzata con diversi algoritmi, la scelta su quale

utilizzare può ricadere su quello che offre un'accuratezza migliore, indipendentemente dalla misura in cui sono soddisfatte le altre proprietà.

Esistono varie tecniche per valutare l'accuratezza di un algoritmo di classificazione. Tra queste le più importanti sono:

- *Metodo holdout.* I dati vengono suddivisi in maniera random tra due insiemi, uno viene utilizzato come *training set* per costruire i modelli, mentre l'altro viene utilizzato come *test set*, per valutare l'accuratezza. Per verificare il livello di accuratezza, si assume che i dati che costituiscono il test set non abbiano etichetta di classe. Si cerca di determinarla utilizzando i modelli definiti dal classificatore: se l'etichetta di classe dell'oggetto appartenente al test set coincide con quello determinato con il classificatore allora la classe per quell'oggetto viene predetta in maniera corretta. Maggiore è il numero di oggetti del test set per cui la classe predetta coincide con la vera classe di appartenenza, maggiore è l'accuratezza fornita dal classificatore. Una buona parte dei dati andranno a costituire il training set, circa i 2/3.
- *Random subsampling.* E' una variante del metodo holdout. Consiste nell'applicare il metodo holdout k volte. Ad ogni iterazione i dati vengono ridistribuiti in maniera random tra *training set* e *test set*. L'accuratezza complessiva è determinata come media delle accuratezze ottenute ad ogni iterazione.
- *K-fold cross validation.* I dati iniziali vengono suddivisi in maniera random tra k sottoinsiemi, t_1, t_2, \dots, t_k . Si effettuano k iterazioni: alla prima iterazione si costruisce il classificatore utilizzando come training set i sottoinsiemi t_2, \dots, t_k e come test set t_1 ; alla seconda iterazione si costruisce il classificatore utilizzando come training set t_1, t_3, \dots, t_k e come test set t_2 , ecc. L'accuratezza complessiva è data dal rapporto tra il numero di classificazioni corrette e il numero totale di tuple considerate.

Tra i metodi elencati sopra, quello più utilizzato è il k-fold cross validation.

3.11 Tecniche di classificazione

Esistono varie tecniche per classificare un insieme di oggetti. Tra queste, le più importanti sono:

- *Classificazione basata su alberi di decisione*. Prevede la costruzione di un albero i cui nodi interni sono condizioni sugli attributi, i nodi foglia rappresentano le classi da predire; un percorso dalla radice ad un nodo foglia descrive le caratteristiche che un oggetto deve avere per appartenere alla classe che etichetta la foglia in quel particolare percorso.
- *Bayesian Classifiers*. E' un classificatore statistico. Si calcola la probabilità che un dato oggetto appartenga ad una classe come probabilità condizionale. Esistono due classificatori bayesiani: *naive bayesian classifier* e *bayesian belief networks*. Il primo assume che l'effetto di un attributo su una data classe sia indipendente dal valore degli altri attributi. Il secondo consente la rappresentazione di dipendenze tra sottoinsiemi di attributi.
- *Classificazione basata su regole associative*. Si cercano pattern frequenti all'interno del training set che possano essere utilizzati per la classificazione.

3.11.1 Alberi di decisione

La tecnica di classificazione scelta in questo lavoro di tesi è quella basata su alberi di decisione. E' il metodo più semplice per classificare degli oggetti in un numero finito di classi. Un algoritmo di classificazione di questo tipo prevede due fasi:

1. Costruzione dell'albero di decisione.
2. Utilizzo dell'albero di decisione costruito per predire la classe di appartenenza di un particolare oggetto.

L'albero di decisione è un diagramma con struttura ad albero, dove i nodi interni rappresentano test sugli attributi e i nodi foglia rappresentano le classi. Lo pseudocodice di un tipico algoritmo per la generazione di un albero di

decisione è riportato in figura 3.7. Si tratta di una versione dell'algoritmo ID3 [Han00].

Algorithm: *Generate_decision_tree*. Generate a decision tree from the given training data.
Input: The training samples, *samples*, represented by discrete-valued attributes; the set of candidate attributes, *attribute-list*.
Output: a decision tree.

Method:

- (1) create a node *N*;
- (2) If *samples* are all of the same class, *C* then
- (3) return *N* as a leaf node labeled with the class *C*;
- (4) If *attribute-list* is empty then
- (5) return *N* as a leaf node labeled with the most common class in *samples*; //majority votino
- (6) Select *test-attribute*, the attribute among *attribute-list* with the highest information gain;
- (7) Label node *N* with *test-attribute*;
- (8) For each known value *ai* of *test-attribute* //partition the samples
- (9) grow a branch from node *N* for the condition *test-attribute=ai* ;
- (10) let *si* be the set of samples in *samples* for which *test-attribute=ai*; //partition
- (11) if *si* is empty then
- (12) attach a leaf labeled with the most common class in *samples*;
- (13) else attach the node returned by ***Generate_decision_tree***(*si*, *attribute-list-test-attribute*);

Figura 3.7 Pseudocodice di un algoritmo per la generazione di un albero di decisione

L'albero iniziale è costituito da un unico nodo che rappresenta il *training sample*. Se le tuple appartengono tutte alla stessa classe, il nodo diventa una foglia ed è etichettato con questa classe; altrimenti l'algoritmo usa una misura basata su entropia detta *information gain* che seleziona l'attributo che meglio separa le tuple in classi individuali. Questo attributo diventa l'attributo "test" o "decision" del nodo. E' l'attributo con information gain massimo, cioè quello che minimizza le informazioni necessarie per classificare le tuple nelle partizioni (sottoalberi) risultanti. Viene creato un ramo per ogni valore conosciuto dell'attributo "test" e le tuple sono suddivise in accordo a questi valori. Si assume che gli attributi siano a valori discreti. Gli attributi a valori continui vanno discretizzati.

Il processo ricorsivo di partizionamento si ferma quando è verificata una delle seguenti condizioni:

1. Tutte le tuple per un dato nodo appartengono alla stessa classe.
2. Non ci sono attributi (tra quelli rimanenti) per i quali si possano fare ulteriori partizioni.
3. Non ci sono tuple per il ramo *test-attribute* = a_i . In questo caso viene creata una foglia etichettata con la classe che occorre più spesso.

Una volta costruito l'albero di decisione, lo si può utilizzare per predire la classe di appartenenza di oggetti privi di etichetta di classe. Si segue un percorso dal nodo radice ad un nodo foglia e ad ogni passo si va in un particolare sottoalbero, se l'arco che porta ad esso è etichettato con una condizione su un attributo soddisfatta dalla tupla da classificare.

Una descrizione dettagliata dell'algoritmo di classificazione utilizzato in questa tesi è fornita in [Rug04].

Esempio 3.3

Una compagnia assicurativa vorrebbe una procedura che sia in grado di segnalare se un nuovo cliente è a rischio. Dispone di un training set, contenente i dati relativi ai propri clienti. Tra gli attributi l'età dell'utente e il tipo di autoveicolo. L'etichetta di classe è il livello di rischio per un particolare soggetto. Per la classe sono previsti due valori: "Alto" e "Basso". L'etichetta di classe è stata assegnata sulla base degli incidenti in cui è stato coinvolto il cliente.

In figura 3.8 è riportato il training set ed il relativo albero di decisione.

	Età	Tipo Autoveicolo	Rischio
Cliente 1	18	Sportiva	Alto
Cliente 2	43	Familiare	Basso
Cliente 3	68	Familiare	Basso
Cliente 4	32	Autocarro	Basso
Cliente 5	23	Familiare	Alto
Cliente 6	18	Familiare	Alto
Cliente 7	20	Familiare	Alto
Cliente 8	45	Sportiva	Alto
Cliente 9	50	Autocarro	Basso
Cliente 10	64	Autocarro	Alto
Cliente 11	46	Familiare	Basso
Cliente 12	40	Familiare	Basso

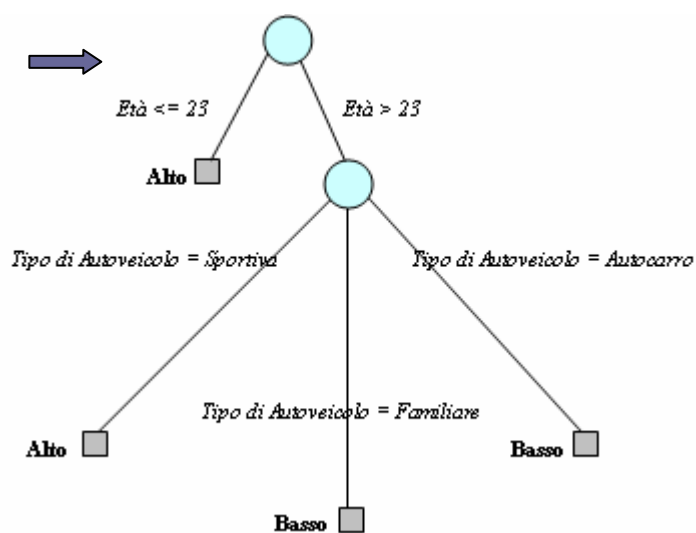


Figura 3.8 Esempio di classificazione mediante un albero di decisione

Supponiamo che ci sia un nuovo potenziale cliente per la compagnia assicurativa, di 45 anni che possiede una macchina di tipo familiare: in base all'albero di decisione il rischio associato a questo soggetto è basso, quindi può essere accettato come nuovo cliente.

3.11.2 Alberi di decisione: Tree Pruning

L'albero di decisione generato a partire da un insieme di dati, può non essere completamente corretto. Nel valutare l'accuratezza di quest'albero, bisogna tener conto dell'errore associato ai dati ed in particolare bisogna prendere in

considerazione la possibilità che tra i dati ci sia “rumore”. Il suo mancato rilevamento si riflette sull'albero e sulla sua accuratezza. I metodi di *pruning* hanno come scopo quello di modificare l'accuratezza dell'albero, per tener conto di rumore che può essere presente nei dati; a tale scopo si eliminano rami dell'albero meno affidabili. Esistono due diversi approcci per realizzare il clustering: approccio *prepruning* e approccio *postpruning*. Nel primo caso, il pruning viene realizzato in fase di costruzione dell'albero, decidendo per esempio dato un nodo di non partizionare ulteriormente un dato insieme di tuple. In questo caso il nodo diventa una foglia.

Con il secondo approccio, il pruning viene realizzato dopo che l'albero è stato generato. In questo caso il pruning consiste nel rimuovere un ramo di un certo nodo.

3.12 Alberi di decisione e regole di classificazione

L'output di un algoritmo di classificazione è un albero le cui foglie sono etichettate con le classi, ogni nodo interno è etichettato con uno tra gli attributi del training set e gli archi uscenti da tale nodo sono etichettati da condizioni sull'attributo mutuamente esclusive. Questo per garantire che in fase di predizione della classe di un determinato oggetto, sia possibile seguire un solo percorso nell'albero. Un percorso dalla radice ad un nodo foglia può essere descritto con una regola del tipo “If..Then..”; la parte “If” della regola è definita come la congiunzione dei vincoli sugli attributi soddisfatti in quel percorso, mentre il ramo “Then” conterrà la classe che etichetta la foglia. Queste regole sono chiamate regole di classificazione; costituiscono una rappresentazione equivalente all'albero di decisione, ma risultano più intuitive. Dato un albero di decisione, si potranno definire tante regole quanti sono i percorsi possibili.

Nel caso ci siano più foglie etichettate con la stessa classe, un oggetto per poter appartenere a questa classe deve rispettare una qualunque delle regole che definiscono il percorso verso queste foglie.

4. La Realtà Virtuale

Esistono in letteratura diverse definizioni di *realtà virtuale*. Tra queste una delle più interessanti definisce la realtà virtuale come *“l’insieme di quelle tecnologie che consentono di ricreare tramite un computer, mondi e oggetti che sono la trasposizione digitale di ambienti reali o di fantasia”* [Vre96]. Questa definizione di realtà virtuale è particolarmente rigorosa; esistono comunque altre definizioni di realtà virtuale altrettanto importanti:

- ❖ *“Un ambiente generato dal computer che simula la realtà in modo che i sensi lo possano percepire”* [Broo99].
- ❖ *“Un’interfaccia uomo-computer in cui un computer crea un ambiente immersivo, che risponde interattivamente, ed è controllato dal comportamento dell’utente”* [Vic04].
- ❖ *“Una simulazione informatica tridimensionale che risponde così realisticamente agli input, da far sembrare di vivere in un altro mondo”* [Fre01].
- ❖ *“Una tecnologia interattiva che controlla totalmente gli input sensoriali e crea la convincente illusione di essere completamente immersi in un mondo generato dal computer”* [Car97].

In base alle definizioni date, un’applicazione di realtà virtuale si può dire tale se:

- implica l’utilizzo di immagini 3D;
- risulta interattiva con l’utente;
- è strutturata in modo che possa prevedere più partecipanti.

Da un punto di vista pratico, per realtà virtuale, s’intende l’uso di un computer per la creazione di un mondo artificiale, tipicamente in tre dimensioni, nel quale l’utente possa muoversi e nel quale possa manipolare degli oggetti.

4.1 Creazione di un ambiente virtuale

Un'applicazione di realtà virtuale crea un ambiente digitale, in cui l'utente ha la possibilità di muoversi in uno spazio tridimensionale in tempo reale, interagendo direttamente con esso. Questo mondo tridimensionale può essere il modello geometrico di un edificio, di un monumento, di una città o di un territorio. Per simulare uno spostamento in tempo reale, il computer calcola istante per istante i dati relativi alla trasformazione della geometria e delle coordinate spaziali delle *textures* che le sono state applicate, secondo i mutamenti di prospettiva e gli spostamenti compiuti dall'utente. Le *textures* sono immagini raster utilizzate per rivestire la geometria, conferendogli i materiali, le proprietà di riflessione, opacità, ecc.

Il processo di creazione di un ambiente virtuale prevede due fasi principali:

- ❖ *Creazione attraverso un programma di grafica 3D* del modello geometrico della realtà che si vuole rappresentare. La geometria è descritta da poligoni (*mesh*), ciascun poligono è individuato da vertici e facce, ciascuna faccia ha una propria normale. E' necessario non superare un certo limite di poligoni, poiché il calcolo del movimento nell'applicazione di realtà virtuale sarà effettuato in tempo reale e modelli troppo pesanti rallenterebbero in maniera eccessiva le prestazioni.
- ❖ *Applicazione di textures al modello geometrico*. Durante questa fase, risulta particolarmente oneroso per il processore e la scheda video il calcolo delle luci in tempo reale. Un ambiente di realtà virtuale, può contenere tre o quattro luci al massimo. Poiché l'illuminazione costituisce un elemento estremamente importante per la resa realistica di un ambiente o di un oggetto, è necessario ricorrere ad alcune strategie per simulare un'illuminazione complessa del modello virtuale, in grado di rispondere a questa esigenza. Una di queste strategie è il *Light Mapping* (Mappa di luce). Il Light Mapping consiste nell'attribuire a ciascuna porzione di geometria

non solo la texture o materiale principale (che ne determina in larga misura l'aspetto), ma anche una texture secondaria in bianco e nero, a cui è affidata l'informazione del chiaroscuro, cioè delle zone di luce e di ombra, che va ad associarsi alla texture principale. Alle zone più chiare corrisponde una maggiore illuminazione, alle più scure l'ombra. Il Light Mapping si ottiene utilizzando appositi programmi di editing sviluppati appositamente per questo scopo.

Un'applicazione di realtà virtuale può essere realizzata con diversi gradi di dettaglio e di precisione a seconda del fine ultimo: può essere realizzata per sistemi di basso livello (computer dalle prestazioni modeste), oppure per sistemi più sofisticati (computer di qualità media e alta) che si avvalgono della capacità di schede video in grado di accelerare il calcolo grafico.

Non è però completamente corretto parlare di computer di bassa, media o alta qualità. In questo settore infatti, la tecnologia si evolve con estrema rapidità: oggi ad esempio, è possibile far funzionare su computer di qualità media applicazioni che fino a due o tre anni fa, erano supportate unicamente da computer di alta qualità.

4.2 Sistemi di Realtà Virtuale: Ambienti immersivi

In base al modo con il quale si interfacciano all'utente, i sistemi di realtà virtuale possono essere suddivisi in:

- *Window on World Systems (WoW) o Desktop VR*: questi sistemi prevedono la visualizzazione del mondo virtuale sul monitor del computer. Lo schermo del computer è paragonabile ad una finestra sul mondo virtuale attraverso la quale è possibile osservarlo. In riferimento a questi sistemi si utilizza il termine "realtà virtuale da scrivania" in quanto, sebbene consenta

un'interazione con ambienti e oggetti sintetici, opera senza l'ausilio di interfacce fisiche particolari, dedicate a rendere più coinvolgente e avvolgente tale esperienza.

- *Video Mapping*: è una variante del WoW. Anche in questo caso lo schermo proietta l'ambiente virtuale; in più l'utente può vedere l'interazione con il mondo virtuale di un soggetto che lo raffigura. Questo viene realizzato con l'ausilio di un sistema di telecamere che consentono di riprendere l'utente e i suoi movimenti nello spazio; si utilizzano poi degli algoritmi complessi per elaborare queste informazioni. L'utente non entra dunque nella simulazione, ma si vede all'interno di essa attraverso lo schermo del computer.
- *Immersive Systems*: l'utente ha la possibilità di navigare il mondo nel modo a lui più congeniale, dalla prospettiva e nella direzione che preferisce. Una maggiore immersione dell'utente è fornita con l'utilizzo di dispositivi particolari, gli Head Mounted Display o l'utilizzo per la proiezione dell'ambiente di una struttura, chiamate CAVE (descritti nel par. 4.4.1).
- *Telepresence*: in questi sistemi l'utente può manovrare a distanza un dispositivo. Uno dei campi di applicazione della telepresence è la chirurgia: per compiere interventi particolarmente delicati, in zone poco accessibili, si possono utilizzare degli strumenti manovrabili a distanza.

Tra questi sistemi di realtà virtuale, quelli più interessanti sono i *sistemi immersivi*. Questi infatti prevedono un'interazione completa dell'utente con l'ambiente virtuale che coinvolge tutti i sensi, non solo la vista, l'udito e il tatto, ma anche quelli che di solito non si è abituati a prendere in considerazione come l'olfatto. La realtà virtuale immersiva prevede l'utilizzo di visori montati in prossimità degli occhi, sensori di posizione in grado di fornire l'indicazione circa l'orientamento dell'utente, guanti dotati di sensori in grado di manipolare gli oggetti virtuali e di simulare sensazioni tattili artificiali. Il livello di immersività fornito dall'ambiente dipende dal tipo di apparecchiature utilizzate per la visione stereo e dalla qualità dei dispositivi di input, dispositivi di output e dispositivi di

visualizzazione. Per avere un buon risultato di immersione, è necessario che i modelli creati siano visualizzati nella maniera corretta.

4.3 Presence

Molti esperimenti di realtà virtuale hanno come scopo quello di valutare il grado di immersione, di presenza, di coinvolgimento dei soggetti sottoposti ad esperimento in relazione al grado di realismo fornito nell'ambiente virtuale ricostruito. Per una trattazione dettagliata sulla relazione tra questi aspetti si rimanda a [Sla01].

La presenza può essere definita come la sensazione di essere realmente nell'ambiente virtuale piuttosto che nel laboratorio nel quale viene condotto l'esperimento e il soggetto si trova fisicamente [San05]. Il concetto di presenza è ben diverso dal concetto di immersione. Il concetto di immersione viene utilizzato per indicare ciò che la tecnologia offre da un punto di vista oggettivo: un ambiente virtuale è tanto più immersivo quanto l'utente può interagire con esso, ha a disposizione gli strumenti per interagire con esso. Il fatto che un ambiente sia immersivo, non significa che l'individuo debba essere necessariamente presente nel mondo virtuale. Un ambiente immersivo può determinare diversi livelli di presenza in individui diversi, così come ambienti virtuali che offrono diversi livelli di immersività possono determinare su un individuo uno stesso livello di presenza. La differenza tra il concetto di immersione e quello di presenza può essere paragonato a quello esistente tra il modo in cui un colore è definito e come viene percepito da una persona. I colori vengono definiti in termini di lunghezza d'onda. Due tonalità di rosso, aventi lunghezze d'onda diverse, possono essere percepite da un soggetto come uno stesso colore. Può succedere inoltre che una certa tonalità di rosso venga percepita diversamente da due soggetti.

Un altro aspetto che viene valutato, è il livello di coinvolgimento di un individuo. Anche questo spesso erroneamente viene utilizzato come sinonimo di presenza. Un individuo può essere presente ma non coinvolto, oppure può essere coinvolto ma non presente (questo accade anche nella vita reale quando per esempio una persona ascolta un concerto di musica classica e legge contemporaneamente un libro). Sebbene i concetti di immersione e coinvolgimento siano ben diversi da quello di presenza, sono comunque aspetti interessanti che vengono considerati congiuntamente a quello di presenza nell'ambito di un esperimento di realtà virtuale.

Il senso di presenza in un ambiente virtuale viene valutato considerando un certo numero di misure, inclusi questionari, misurazioni fisiologiche e i breaks in presence riportati durante l'esperimento.

4.3.1 Breaks in presence

Un break in presence (BIP) occorre quando l'individuo risponde più agli stimoli del mondo reale che a quelli del mondo virtuale. Un individuo è presente nel mondo virtuale quando risponde agli stimoli provenienti da esso.

Infatti chi partecipa ad un esperimento di realtà virtuale è soggetto a due flussi di stimoli, quelli provenienti dal mondo reale e quelli provenienti dal mondo virtuale. Un break in presence occorre quando il partecipante smette di rispondere agli stimoli provenienti dal mondo virtuale e risponde invece agli stimoli provenienti dalla realtà.

In generale i BIPs vengono rilevati in due modi: durante l'esperimento o al termine. Ovviamente la rilevazione durante l'esperimento è più precisa: l'individuo ha a disposizione un dispositivo dotato di pulsante: ogni qual volta egli sente la realtà predominare sull'esperimento deve premere il pulsante.

Alcuni studiosi si limitano a rilevare la presenza dopo l'esperienza immersiva mediante un questionario. In alcuni casi questo non è sufficiente in quanto può essere utile conoscere gli istanti di tempo in cui il soggetto non è presente e la

relazione tra *BIPs* e variazione dei parametri fisiologici negli istanti in cui occorrono. Spesso si usano congiuntamente *BIPs* e questionari per avere informazioni più precise sulla presenza.

4.3.2 Parametri fisiologici

La misurazione dei parametri fisiologici nasce dall'esigenza di avere informazioni affidabili circa l'esperienza di un individuo in un ambiente virtuale.

Gli esperti di realtà virtuale utilizzano le misurazioni dei parametri fisiologici per verificare quanto è stata credibile l'esperienza dei partecipanti. L'idea è che se il partecipante era presente nell'ambiente, allora avrà risposte fisiologiche simili a quelle che avrebbe nel mondo reale. I parametri fisiologici che vengono misurati sono l'attività elettrica del cuore (elettrocardiogramma), la respirazione e la sudorazione. Perché queste misurazioni siano significative, si tiene conto anche dei valori in una situazione normale; si analizza quindi la variazione di questi parametri.

L'elettrocardiogramma (ECG) fornisce una rappresentazione dell'attività elettrica del cuore. Da un'analisi dell'elettrocardiogramma è possibile estrarre diverse informazioni:

- *Heart rate* cioè il numero di contrazioni per minuto.
- *RR intervals* che indicano la distanza nel tempo tra due contrazioni consecutive.
- *Heart Rate Variability (HRV)* cioè la variazione nel tempo della frequenza cardiaca.

Il GSR (Galvanic skin response) misura il cambiamento nella resistenza elettrica cutanea. Viene determinata facendo passare una corrente debole attraverso la pelle o misurando la tensione generata dal corpo stesso. E' strettamente correlata con l'emozione, l'attenzione e lo stress. Il GSR è legato

positivamente con l'ansia di un soggetto; può essere utilizzata per esempio nelle simulazioni di volo per valutarne il livello.

Alcuni studi condotti nell'ambito della realtà virtuale hanno dimostrato che la sudorazione e l'heart rate sono in stretta correlazione con i BIPs; da un'analisi dei parametri fisiologici in un intorno dell'istante in cui il partecipante segnala il BIP, si rileva una variazione significativa di questi parametri. In particolare risulta che l'heart rate raggiunge un picco approssimativamente un secondo prima che occorra il BIP. La variazione nell'heart rate può essere dovuta all'azione o all'intenzione di premere il pulsante per segnalare il BIP più che al BIP in se stesso. Nel caso in cui sia stata misurata la variazione della sudorazione è stato rilevato un aumento della sudorazione circa 1,8 secondi dopo che il soggetto preme il pulsante per segnalare il BIP.

Per maggiori dettagli relativi alla relazione esistente tra BIPs e parametri fisiologici si rimanda a [Sla03].

4.3.3 Questionari SUS e ITC-SOPI

Spesso per avere una stima della presenza di un soggetto in un ambiente virtuale gli si chiede di rispondere ad un questionario. Secondo alcuni studiosi (vedi [IJ00]) l'analisi della presenza in un ambiente virtuale fatta utilizzando solo i questionari non è sufficiente, soprattutto perché non si hanno informazioni temporali su quando un soggetto non è presente. A tale scopo è stata introdotta la misurazione dei parametri fisiologici. I questionari sono misurazioni soggettive, diversamente dai parametri fisiologici che sono misurazioni oggettive.

Il vantaggio che nasce dall'uso dei questionari è il fatto che sottoponendoli dopo l'esperimento, non si influenza l'esperimento ed è facile gestire i dati che ne derivano. Il principale svantaggio è l'impossibilità di avere informazioni temporali relativamente agli istanti in cui il soggetto non è presente.

4.4 Interazione in Realtà virtuale: una panoramica generale

La creazione di un ambiente virtuale, comporta la progettazione di una macchina che simuli stimoli sensoriali, visivi e sonori, in modo da dare all'utente la sensazione di essere realmente in uno spazio differente da quello in cui si trova.

Questo implica lo studio di efficienti metodi di interazione tra l'utente e il mondo virtuale e quindi lo studio di sistemi di navigazione, strumenti per la visione tridimensionale e strumenti per la manipolazione di oggetti virtuali.

4.4.1 Strumenti per la visione tridimensionale

Gli oggetti di un ambiente virtuale sono tridimensionali. Poiché l'ambiente è visualizzato su uno schermo, le coordinate tridimensionali vengono convertite in coordinate bidimensionali. Perché l'utente percepisca la tridimensionalità degli oggetti, sono necessari particolari strumenti che vengono sistemati in prossimità degli occhi e consentono di avere tale visione tridimensionale. Questi strumenti sfruttano le caratteristiche dell'occhio umano. Gli occhi di un essere umano sono distanziati di circa 7-8 cm, per cui quando si osserva un'immagine, quella prodotta dall'occhio sinistro, risulta leggermente sfasata rispetto a quella prodotta dall'occhio destro. Il cervello riceve quindi due immagini, le rielabora aggiungendo la dimensione mancante: la profondità. Le speciali apparecchiature utilizzate per la navigazione di un ambiente virtuale, utilizzano questo principio e attraverso differenti tipi di tecniche, fanno ricevere alla retina due immagini sfalsate. Tra queste apparecchiature, di particolare interesse sono gli occhiali stereoscopici a lenti colorate o polarizzate. Con questi occhiali è possibile ottenere una visione tridimensionale, fotografando un oggetto due volte con una posizione sfasata di 7-8 cm, utilizzando per le immagini due filtri con colori complementari (ad esempio rosso e blu). Le due

fotografie vengono poi fuse insieme ottenendo una foto tridimensionale visualizzabile con degli occhiali aventi una lente per colore.

L'utilizzo di questa tecnica produce risultati scadenti con immagini a colori. Per ovviare a questo problema, è possibile sostituire ai filtri e alle lenti colorate, dei filtri e delle lenti polarizzate.

Particolarmente interessanti sono anche gli occhiali stereoscopici ad otturatori, con i quali si può ottenere una visualizzazione stereoscopica utilizzando solamente lo schermo del computer. La visualizzazione stereoscopica, è possibile grazie alla visualizzazione contemporanea di due immagini, una per occhio. Gli occhiali ad otturatori montano, al posto delle lenti, dei filtri a cristalli liquidi che, se polarizzati correttamente, diventano completamente neri, impedendo all'occhio di vedere attraverso essi. Le immagini vengono montate in sequenza sullo schermo alternando ripetutamente i frames destinati all'occhio destro con quelli destinati all'occhio sinistro. In contemporanea, viene inviato un segnale agli occhiali in modo da oscurare l'occhio non interessato dall'immagine presente sullo schermo. Questo implica la necessità di utilizzare macchine molto veloci, in quanto per ottenere 25 fotogrammi al secondo, è necessario visualizzare nello stesso periodo di tempo 50 immagini (25 per ogni occhio). Il vantaggio di questa tecnologia, è rappresentato dall'alta definizione grafica ottenibile, che dipende solamente dalla velocità del sistema di elaborazione.

Un altro dispositivo per la visualizzazione è rappresentato dal HMD (Head Mounted Display). Si tratta di un casco che contiene un display per la visualizzazione delle immagini. Al suo interno sono integrate delle cuffie stereofoniche di alta qualità per permettere l'utilizzo di audio tridimensionale. Le tecniche usate per visualizzare le immagini in un HMD, dipendono dalle funzionalità offerte dallo strumento e dalla qualità dell'apparecchiatura. Gli HMD possono essere stereoscopici, quando esiste un visore diverso per

occhio, o monoscopici, quando entrambi gli occhi visualizzano la stessa immagine presente su un singolo display.

Gli HMD di costo medio utilizzano dei mini schermi a cristalli liquidi che, allo stato attuale della tecnologia, permettono una risoluzione massima di 768 x 512 pixel. Sono comunque allo studio, display a cristalli liquidi in grado di visualizzare 1024 x 768 punti con 16 milioni di colori. Gli HMD di questo tipo sono ancora allo stadio prototipo; sono utilizzati solamente in applicazioni critiche (ad esempio simulazioni militari). Il vantaggio dell'utilizzo di display a cristalli liquidi, è dato dalle ridotte dimensioni degli HMD che li montano e dal fatto che essi non emettono radiazioni elettromagnetiche.

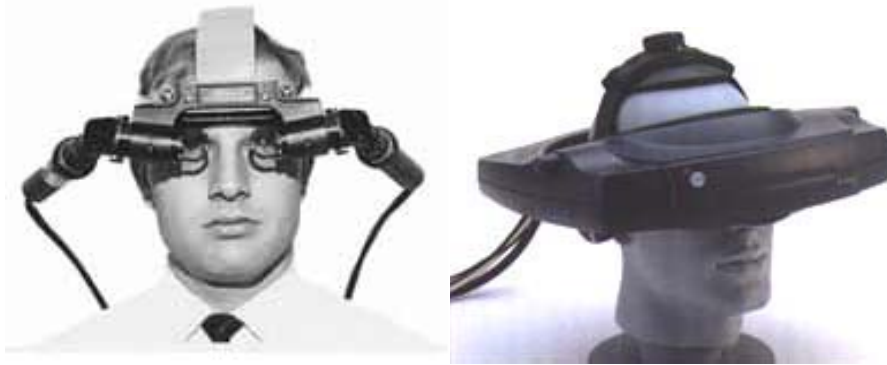


Figura 4.1 (a) Il primo HMD

(b) HMD moderno

Un altro modo per dare agli utenti una visione tridimensionale degli oggetti virtuali, è rappresentata dall'utilizzo di una struttura particolare, detta CAVE: si tratta di una struttura in cui si entra fisicamente ovvero una stanza sulle cui pareti viene proiettato l'ambiente virtuale. Il numero di pareti in cui viene visualizzato l'ambiente varia da tre a sei.

I principali vantaggi derivanti dall'utilizzo di questa struttura sono:

- un vasto campo di vista;
- la possibilità di far condividere una stessa esperienza ad un gruppo di soggetti.

I principali svantaggi sono:

- il costo eccessivo dei sistemi di generazione di immagini multiple;
- la necessità di spazio.

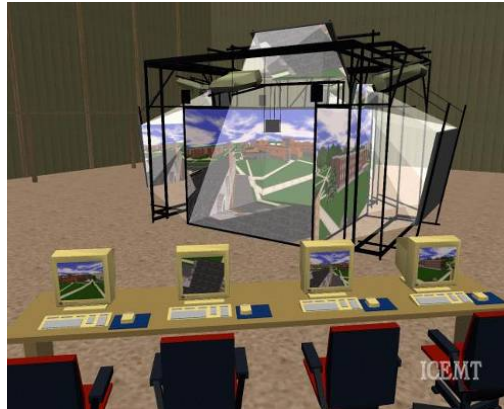


Figura 4.2 Cave

4.4.2 Sistemi di navigazione e strumenti per la manipolazione di oggetti

L'utente deve poter interagire con l'ambiente virtuale, deve poter in qualche modo "navigare" nell'ambiente virtuale, inviare dei comandi che gli consentano di muoversi in una certa direzione o di compiere una certa azione. Esistono diversi dispositivi che consentono la navigazione e l'interazione dell'utente con l'ambiente. Tra questi quello più utilizzato, è il comune joystick, un semplice dispositivo dotato di pulsanti. L'utente preme un pulsante diverso a seconda dell'azione che vuole compiere. All'azione dell'utente corrisponde una reazione del calcolatore. Esistono altri dispositivi che devono essere indossati dall'utente: questi sono in grado di interpretare il linguaggio del corpo, di tradurlo. Attualmente sono in fase di sviluppo dispositivi in grado di interpretare anche semplici messaggi vocali.



Figura 4.3. Manipolazione di oggetti: data gloves

Tra gli strumenti che consentono di interagire con l'ambiente virtuale, particolarmente interessante è il *data glove*, uno speciale guanto che consente di manipolare oggetti virtuali. Il *data glove* consente anche di rilevare la posizione dell'utente; è dotato infatti di sensori in grado di intercettare la posizione e il movimento di ogni singolo dito: dalla posizione nello spazio della mano, è possibile determinare la posizione nello spazio del soggetto.

I *data gloves* più sofisticati sono in grado di fornire una risposta tattile alle azioni dell'utente. Per far ciò utilizzano delle piccole scariche elettriche per stimolare le connessioni nervose sottocutanee oppure utilizzano dei materiali in grado di flettersi quando sono sottoposti all'azione di un campo elettrico. In questo modo è possibile fornire all'utente la sensazione di impugnare realmente un oggetto.

4.5 Sistemi per il rilevamento della posizione in VR

Un esperimento virtuale prevede il coinvolgimento di individui che, dotati di speciali apparecchiature, si muovono in un ambiente artificiale. Grazie a speciali dispositivi, è possibile determinare in ogni istante la posizione di un soggetto. Questo permette di fare poi delle analisi più approfondite circa la

traiettorie dei singoli soggetti e le relazioni esistenti tra esse, le misurazioni fisiologiche e il grado di presenza dei soggetti stessi.

In base alle tecnologie utilizzate, i sistemi per tracciare il movimento di un individuo, all'interno di un mondo virtuale, si suddividono in:

- *Meccanici*: legano fisicamente l'utente alla macchina mediante un braccio snodato, nelle giunture del quale sono presenti dei sensori che rilevano gli spostamenti. Sono ingombranti e scomodi da usare, ma permettono un'elevata velocità operativa. La loro applicazione è confinata a sistemi di teleoperazione e guida a distanza (ad esempio il controllo a distanza del braccio meccanico di un robot).
- *Ottici*: si basano sull'utilizzo di telecamere che rilevano la posizione di sorgenti di luce poste sull'operatore, ed eseguono i calcoli sullo spostamento basandosi sull'elaborazione in tempo reale delle immagini acquisite.
- *Ultrasonici*: sfruttano la propagazione delle onde sonore, si basano sulla rilevazione della riflessione delle onde e della deviazione che esse subiscono quando un oggetto si muove al loro interno.
- *Magnetici*: sono i sistemi più utilizzati. Una sorgente fissa genera un campo magnetico e l'operatore indossa dei dispositivi in grado di provocare distorsioni all'interno di esso. La posizione dell'operatore viene determinata da dei sensori che rilevano le variazioni dei campi magnetici.

4.6 Un linguaggio per costruire ambienti virtuali: VRML

VRML (Virtual Reality Modeling Language) è un linguaggio di programmazione che consente la simulazione di mondi virtuali tridimensionali. Mediante il VRML è possibile descrivere ambienti virtuali contenenti oggetti, sorgenti luminose, immagini, suoni, filmati. E' nato dall'esigenza di creare ambienti virtuali 3D

accessibili tramite internet, così che un utente web possa esplorare un mondo virtuale così come esplora il testo e la grafica di un sito. Esistono due versioni del VRML, 1.0 e 2.0: la prima consente di descrivere mondi statici, mentre la seconda consente di definire la dinamica degli oggetti virtuali. In particolare VRML 2.0 ha funzionalità che consentono l'introduzione di animazioni e interazioni con l'utente, che sono le caratteristiche indispensabili per un'applicazione di realtà virtuale. Un file vrml, avente estensione .wrl, è un file ascii che contiene tutti i comandi necessari per descrivere il mondo virtuale; in particolare contiene comandi che servono a descrivere la forma dei singoli oggetti, il loro colore, le textures, ecc. e a posizionare tali oggetti in uno spazio 3D.

VRML potrebbe essere considerato simile ad HTML: un file VRML pubblicato su internet è accessibile da qualsiasi macchina indipendentemente dalla piattaforma, nello stesso modo in cui i documenti HTML possono essere visualizzati sostanzialmente allo stesso modo su macchine Windows, Mac, Unix, Linux, ecc. VRML risulta particolarmente utile per progetti che richiedono massima interazione, animazione, la partecipazione dell'utente e l'esplorazione oltre alla visualizzazione statica di testi e immagini di cui HTML è già provvisto. Si pensi ad esempio all'utilità di questo strumento per una concessionaria che voglia commercializzare motocicli. L'azienda rivenditrice potrebbe utilizzare un normalissimo sito web: utilizzando l'HTML, è possibile realizzare un sito dall'interfaccia grafica molto curata, inserire immagini dettagliate dei vari modelli, pubblicare video-clip che mostrino i vari modelli in funzione. Utilizzando Flash, le possibilità a disposizione del Web Designer aumentano sensibilmente, ma in ogni caso il risultato finale rimane per lo più distante dall'esperienza naturale di vivere in un mondo tridimensionale. Con VRML è possibile offrire all'utente la sensazione di entrare in uno spazio 3D dove è presente il modello della moto selezionata, camminarvi attorno e osservarla sotto diversi punti di vista, persino guidarla lungo strade virtuali. Molte di queste cose sono in effetti

già realizzabili in altri modi (si pensi a QuickTime VR, agli applet Java, ecc), ma in ogni caso le alternative a VRML per ora non raggiungono il medesimo livello di interattività e di universalità.

Il principale vantaggio che nasce dall'uso di VRML, è la possibilità di realizzare effettivamente un ambiente interattivo, in cui le azioni dell'utente hanno su di esso degli effetti visibili in tempo reale. Ci sono però anche degli aspetti negativi: in particolare il fatto che si cerchi di interagire con un mondo 3D avendo a disposizione degli strumenti di navigazione che sono 2D. Spostarsi all'interno di una pagina HTML utilizzando un mouse è immediato; richiede invece un po' più di tempo imparare a navigare in uno spazio 3D con una periferica nata per il 2D.

In questi ultimi anni il VRML sta trovando crescente diffusione e uno degli obiettivi è quello di estendere il linguaggio in modo da poter definire ambienti virtuali condivisibili, nei quali più utenti connessi possono interagire tra loro in diverso modo (con la rappresentazione fisica dell'utente all'interno dell'ambiente, chiamata avatar). La realizzazione di tali ambienti è già possibile, sia attraverso l'utilizzo di particolari applicazioni esterne che sfruttano VRML, sia attraverso l'interfaccia VRML verso Java (sottoposta a standardizzazione).

4.7 Realtà virtuale e Computer Graphics a confronto

Spesso si tende a confondere la realtà virtuale con la computer graphics ; in realtà pur essendo strettamente legate, sono discipline diverse.

La principale differenza è che nella computer graphics, si dà molto peso alla qualità visiva delle immagini. Esistono algoritmi di *rendering* e *ray tracing* molto complessi, per elaborare le immagini; le animazioni si ottengono con l'utilizzo di più fotogrammi e per ottenere una buona qualità grafica il tutto è calcolato in maniera preventiva.

Nella realtà virtuale, invece, le immagini non sono calcolate in precedenza, ma vengono generate in tempo reale. Questo è dovuto al fatto che l'ambiente virtuale è caratterizzato, come abbiamo già accennato in precedenza, da una forte interazione tra utente e ambiente. Nelle animazioni tradizionali, invece, l'utente può solo decidere quale parte della sequenza di fotogrammi visualizzare; può spostarsi in avanti, ritornare all'inizio, decidere la velocità di scorrimento, ma ciò che vede è quello che è stato programmato in precedenza. Nella realtà virtuale, è l'utente che decide quello che vuole vedere, può scegliere il punto di vista a lui più congeniale; può posizionarsi in un qualsiasi punto dello spazio, può selezionare un oggetto, ruotarlo, cambiarne le dimensioni, i colori e se abilitato, può interagire con tutto l'ambiente circostante. Per ottenere un effetto sufficientemente fluido, è necessario visualizzare almeno venticinque fotogrammi al secondo; in caso contrario, i movimenti nel mondo virtuale risultano essere imprecisi e a scatti. Per questo motivo, la qualità delle immagini, è lontana dalla qualità fotografica ottenuta con le tecniche di animazione digitale. Due o più individui possono navigare contemporaneamente in un ambiente virtuale: ognuno interagirà con l'ambiente indipendentemente dagli altri e ognuno avrà la propria visione dell'ambiente. Ad esempio, si supponga che A e B siano due attori che si trovano nell'ambiente virtuale: se A decide di aprire una porta (virtuale) nello stesso istante di tempo in cui B decide di chiuderla, il sistema dovrà mostrare ad A una porta aperta e a B una porta chiusa. L'effetto è diverso da quello che si avrebbe nella realtà, ovvero è incoerente rispetto alla realtà.

4.8 Realtà virtuale: applicazioni

La realtà virtuale è utilizzata in vari campi, che si estendono dall'educazione alla visualizzazione spaziale, dall'addestramento alla visualizzazione scientifica, dai beni culturali all'intrattenimento.

Nei paesi tecnologicamente avanzati, viene utilizzata la simulazione virtuale per l'addestramento militare di piloti e sommergibilisti alle prime armi. L'addestramento virtuale, consente alla persona da addestrare, di acquisire rapidamente la pratica nelle operazioni, portando avanti la maggior parte dell'addestramento nel simulatore interattivo, dove sia l'ambiente dell'operazione che gli strumenti impiegati sono ricreati virtualmente. Questo approccio, si adatta particolarmente a operazioni che se condotte nel mondo reale, esporrebbero l'operatore ancora inesperto a rischi, dove una manovra non appropriata potrebbe condurre a situazioni pericolose per la persona o a danni alle strutture fisiche. Oltre ad una riduzione dei rischi a cui sono sottoposti i piloti, c'è anche una riduzione sui costi: è possibile simulare piattaforme diverse con poche modifiche.

Per quel che riguarda i beni culturali, la navigazione virtuale può essere utilizzata per accedere a monumenti non aperti al pubblico o a rischio danneggiamento; è possibile ricostruire scenari o paesaggi non più esistenti, navigare senza limiti di spazio o di tempo, fruire di oggetti o collezioni altrimenti impossibili. Nell'ambito artistico, la realtà virtuale offre un'ottima assistenza in fase di restauro di opere e monumenti; Può essere anche essa stessa uno strumento per creare opere d'arte.

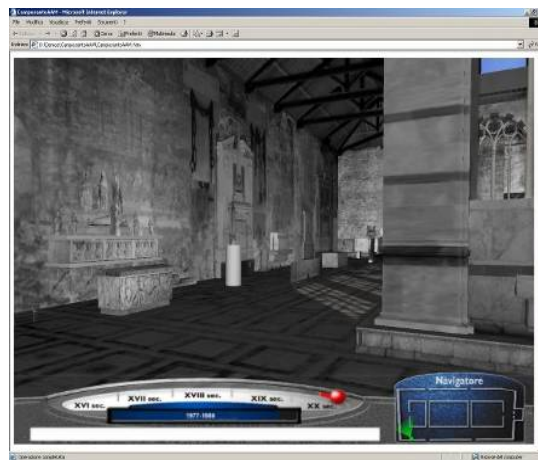


Figura 4.4 Applicazione della RV ai beni culturali

Un altro campo di applicazione della realtà virtuale è rappresentato dall'istruzione: costituisce uno strumento di supporto per gli insegnanti essendo una tecnologia di grande impatto e quindi particolarmente stimolante e coinvolgente. Aiuta ad apprendere concetti nuovi in maniera interattiva, cosa che facilita l'apprendimento e lo rende più efficace.

La realtà virtuale risulta particolarmente utile nell'ambito medico: è possibile infatti simulare un'operazione chirurgica, non solo a fini di addestramento, ma anche per pianificare un'operazione reale sulla base dei dati fisicamente rilevati sul paziente. Oltre che nella chirurgia, è possibile utilizzarla per addestrare i medici sulla palpazione dei tessuti, l'inserzione di aghi, ecc.



Figura 4.5 Applicazione della RV in campo medico

Le applicazioni di realtà virtuale sono molto utilizzate anche nell'ambito dell'architettura: in questo caso gli utenti possono esplorare in tempo reale una scena 3D che rappresenta un ambiente architettonico, possono quindi valutare gli spazi, l'illuminazione, i materiali, l'acustica. In questo ambito, la realtà virtuale è un ottimo strumento di modellazione per analizzare gli spazi dall'interno e valutare differenti scelte di progetto.



Figura 4.6 Utilizzo della RV per gli studi architettonici

La realtà virtuale viene utilizzata anche nell'ambito produttivo. Tra i vantaggi in questo caso, c'è una generale riduzione di tempi e di costi di progettazione dei prodotti e un corrispondente aumento della qualità principalmente per la capacità di valutare e provare approfonditamente un prodotto fin da uno stadio preliminare della sua progettazione in un ambiente virtuale altamente interattivo e facilmente riconfigurabile.

Uno dei più redditizi campi applicativi è l'intrattenimento. L'industria videoludica ha dato un'enorme spinta tecnologica che ha prodotto aumenti di performance e riduzioni dei costi. I simulatori, oltre che utili per il training, sono strumenti di intrattenimento molto attraenti.

5. Case study: EQUATOR

Lo scopo di questo lavoro di tesi è verificare l'applicabilità del data mining al campo della realtà virtuale. Tale verifica è stata condotta applicando tecniche di data mining ai dati relativi ad un esperimento condotto presso il dipartimento di informatica della *University College London*. Lo scopo di questo capitolo è quello di descrivere questo esperimento nel dettaglio per mettere in evidenza i dati e le informazioni che sono risultati utili per la nostra analisi. In questo capitolo ci si limita a descrivere quindi lo scopo per il quale è stato condotto questo esperimento, l'ambiente virtuale, le informazioni e i dati estratti. Nel capitolo 6 è descritto come questi dati sono stati analizzati e quali sono stati i risultati dell'applicazione delle tecniche di data mining.

5.1 EQUATOR

L'esperimento di realtà virtuale oggetto del nostro studio, s'inserisce all'interno di un progetto denominato *EQUATOR* che ha come scopo quello di studiare l'interazione tra individui e mondo virtuale. Per una descrizione dettagliata dell'esperimento si rimanda a [Vina04].

L'obiettivo dell'esperimento specifico è quello di valutare gli effetti della qualità visiva dell'ambiente sui soggetti sottoposti all'esperimento.

L'esperimento ha coinvolto quaranta persone, di cui una metà di sesso femminile e l'altra di sesso maschile.

Sono stati condotti quattro singoli esperimenti nello stesso ambiente virtuale, variando solo la qualità grafica delle textures applicate agli edifici e la qualità grafica degli avatars che popolano l'ambiente; i quaranta soggetti sono stati

suddivisi in maniera casuale in quattro gruppi di dieci persone e bilanciati per sesso.

L'ambiente ricreato per ciascuno dei quattro gruppi di partecipanti è stato uno dei seguenti:

- Ambiente popolato da avatar realistici e qualità grafica alta degli edifici.
- Ambiente popolato da avatar realistici e qualità grafica bassa degli edifici.
- Ambiente popolato da avatar cartoon-like e qualità grafica alta degli edifici.
- Ambiente popolato da avatar cartoon-like e qualità grafica bassa degli edifici.

La qualità grafica degli edifici è definita in funzione della texture, ossia del materiale applicato alle insegne dei negozi. La qualità è bassa se il numero di textures diverse utilizzate è basso (circa 20); la qualità è alta se il numero di textures diverse utilizzate è alto (circa 40).

Nella tabella 5.1 è riportata una descrizione sommaria della suddivisione dei partecipanti nei quattro ambienti.

	<i>Cartoon Form Characters</i>	<i>Higher fidelity Characters</i>
<i>Repetitive Textures</i>	5 persone di sesso femminile 5 persone di sesso maschile	5 persone di sesso femminile 5 persone di sesso maschile
<i>Non-Repetitive Textures</i>	5 persone di sesso femminile 5 persone di sesso maschile	5 persone di sesso femminile 5 persone di sesso maschile

Tabella 5.1 Caratteristiche degli esperimenti condotti.

5.1.1 L'ambiente virtuale

L'ambiente virtuale, in cui è stato condotto l'esperimento, è la ricostruzione di un centro commerciale rappresentato da una strada ai cui lati sono ubicati una serie di negozi.



Figura 5.1 Centro commerciale "virtuale"

I quattro esperimenti condotti si differenziano per la qualità degli avatars che popolano l'ambiente e per la qualità grafica delle insegne dei negozi. Maggiore è il numero di textures diverse utilizzate, maggiore è la qualità grafica dell'ambiente. Per fornire all'utente una visione tridimensionale è stata utilizzata una struttura particolare simile al CAVE [Cru93], chiamata REACTOR. Si tratta di una struttura in cui si entra fisicamente, una stanza alle cui pareti è proiettato il mondo virtuale. Maggiori dettagli su questa struttura sono forniti nel capitolo 4. I soggetti sottoposti all'esperimento erano provvisti di occhiali particolari per la navigazione e di un joystick per segnalare i breaks in presence e per scegliere la direzione nella quale muoversi.

5.1.2 Avatars

L'ambiente virtuale è popolato da *avatars*, cioè personaggi virtuali che si muovono nello spazio artificiale ricreato, animati con l'utilizzo di particolari algoritmi. L'algoritmo utilizzato per l'animazione è in grado di rilevare la posizione dell'avatar, nonché la posizione di eventuali ostacoli e quindi degli altri avatars, dei partecipanti all'esperimento, dei negozi ecc. L'algoritmo tiene traccia della posizioni di tutti gli oggetti presenti nell'ambiente virtuale; nel caso in cui rilevi una collisione, presente quando la distanza tra due oggetti è inferiore ad una certa soglia, modifica la traiettoria dell'avatar di cui definisce il percorso. Gli avatars sono animati con lo stesso algoritmo, quindi compiono gli stessi movimenti. Per rendere più realistica la scena, gli avatars iniziano a muoversi nell'ambiente virtuale in istanti di tempo differenti; in questo modo non risultano più sincronizzati e apparentemente compiono movimenti diversi.

Poiché lo scopo dell'esperimento era quello di studiare la necessità di coerenza nella qualità grafica dei vari elementi virtuali, sono stati utilizzati due tipi di avatars:

- Avatar realistici (*Higher Fidelity Characters*): si tratta di individui virtuali che hanno sembianze umane; è difficile definire avatar di questo tipo, essendo la realtà molto complessa e difficile da modellare. Sono tanto più realistici quanto più un utente ha difficoltà a distinguere una persona reale da una persona virtuale.
- Avatar non realistici (*Cartoon Form Characters*): si tratta di individui virtuali poco realistici che mettono in evidenza la natura dell'ambiente virtuale.



Figura 5.2 (a) Higher Fidelity Characters



Figura 5.2 (b) Cartoon Form Characters

Esistono in letteratura diversi studi riguardanti la qualità degli avatars e come questa può influenzare la credibilità dell'ambiente virtuale. Per una trattazione dettagliata sull'argomento si rimanda a [Fre00, Gar03]. E' opinione comune che avatars realistici rendano l'ambiente credibile. Tuttavia, riprodurre un individuo reale non è una cosa semplice; la realtà è molto complessa e trascurare particolari apparentemente insignificanti può produrre risultati non buoni. Spesso, per rendere un avatars realistico, lo si implementa studiando anche un movimento degli occhi. Un altro elemento che rende realistico un avatar è la capacità di rispondere a stimoli esterni; è previsto in questo caso lo studio di reazioni da implementare negli avatars, eseguite in risposta ad azioni compiute da uno dei soggetti sottoposti all'esperimento.

5.1.3 Fasi dell'esperimento

Ciascuno dei 4 esperimenti condotti si articola in 4 fasi fondamentali:

- *Fase preliminare.* Durante questa fase ogni partecipante ha risposto ad una serie di questionari; lo scopo era quello di acquisire le informazioni demografiche. Una volta terminata la compilazione del questionario, gli sperimentatori hanno spiegato ai partecipanti il concetto di *break in presence*. Si ha un *break in presence* quando l'utente, che naviga nel

mondo virtuale, sente prevalere il mondo reale (sente ad esempio gli stimoli che provengono dal laboratorio) rispetto a quello virtuale. Per spiegare il concetto è stata utilizzata la tecnica di Gestald. Questa tecnica prevede l'utilizzo di immagini che hanno un doppio significato visivo a seconda di come vengono osservate. I partecipanti dovevano osservare le immagini spostando l'attenzione sui due contenuti / significati dell'immagine. Lo stacco percepito nello spostare l'attenzione da un contenuto all'altro poteva essere paragonato alla sensazione provata nel passare dal mondo virtuale a quello reale o viceversa.

- *Baseline.* Durante questa fase ciascun partecipante è stato equipaggiato con speciali apparecchiature, per rilevare i parametri fisiologici, cioè sudorazione, respirazione ed attività cardiaca. Già durante questa fase sono stati misurati questi parametri. Lo scopo era quello di confrontarli con quelli rilevati nella fase successiva, durante l'esperimento vero e proprio. Gli esperti di realtà virtuale hanno quindi studiato la variazione che si è avuta durante l'esperimento, nei parametri fisiologici dei soggetti. Questa fase ha avuto una durata di circa un minuto e mezzo.
- *Fase di training.* Durante questa fase gli sperimentatori hanno mostrato ai partecipanti le modalità per muoversi nell'ambiente. Anche questa fase ha avuto una durata di pochi minuti.
- *Fase di navigazione.* Si tratta dell'esperimento virtuale vero e proprio. I partecipanti sono stati invitati a navigare nell'ambiente virtuale. Il loro compito era quello di muoversi nel centro commerciale virtuale tra i negozi e segnalare attraverso un joystick i breaks in presence. La durata dell'esperimento è stata di tre minuti per ciascun partecipante.
- *Fase post-esperimento.* Durante questa fase, a ciascun partecipante è stato chiesto di rispondere ad un questionario riguardante il livello di presenza. I risultati di questo questionario sono stati utilizzati dagli sperimentatori per

analizzare i breaks in presence riportati da ciascun partecipante durante l'esperimento.

5.1.4 Risultati

Lo scopo degli esperti di realtà virtuale era quello di verificare la necessità di un certo livello di consistenza tra la qualità grafica degli elementi dell'ambiente virtuale. Questo studio richiede di valutare il legame tra presenza riportata dai partecipanti e qualità grafica delle textures applicate agli edifici; richiede inoltre di valutare il legame tra la qualità degli avatar e la presenza riportata. Infine richiede lo studio della necessità di consistenza tra i livelli di realismo dei vari elementi della scena. Da un'analisi dei breaks in presence e del questionario post-esperimento è risultato che si è registrata minore presenza nel caso in cui gli avatars siano realistici e la qualità grafica sia bassa. Questo risultato può essere dovuto al fatto che i partecipanti possono aver rilevato l'inconsistenza tra i vari elementi della scena; possono quindi aver percepito la natura poco realistica dell'ambiente. Poiché nel caso di avatar realistici e numero di textures alto non si ha il minimo numero di break in presence, l'ipotesi sul fatto che sia necessaria una certa consistenza tra i vari elementi della scena non è pienamente soddisfatta.

6 Analisi

Il lavoro di tesi ha avuto come scopo principale quello di verificare l'applicabilità del data mining al campo della realtà virtuale. Non è descritto in letteratura nessun utilizzo delle tecniche di data mining su dati di questo tipo; sarebbe interessante verificare se una tale applicazione sia possibile e possa fornire risultati interessanti.

In questo modo si raggiungerebbe un duplice successo: da un lato si troverebbe un nuovo campo applicativo per il data mining, accrescendone così l'importanza come potente strumento per l'analisi di grosse moli di dati; dall'altro, le tecniche di data mining rappresenterebbero una valida alternativa ai metodi statistici tradizionali utilizzati dagli esperti di realtà virtuale sui dati estratti durante un esperimento di questo tipo.

Lo scopo di questo lavoro di tesi è quindi quello di verificare se queste discipline molto diverse possano essere considerate complementari.

Per realizzare questo lavoro siamo partiti dai dati di un esperimento di realtà virtuale (descritto nel capitolo 5) condotto presso il dipartimento di informatica della *University College London*. L'analisi non ha coinvolto tutti i dati che ci sono stati forniti ma solo una parte di essi, quelli che dal nostro punto di vista potevano risultare più utili e interessanti.

Nei paragrafi successivi è fornita una descrizione dei dati a cui si è ristretta la nostra analisi. Viene fornita anche una descrizione dell'approccio seguito per trattare questi dati con il clustering e la classificazione; viene spiegato inoltre il perché della scelta di queste tecniche di estrazione di conoscenza a scapito di altre tecniche esistenti.

6.1 I dati

L'esperimento oggetto del nostro studio ha prodotto una grande quantità di informazioni in cui è possibile individuare diversi tipi di dato:

- Dati che descrivono l'ambiente virtuale;
- Dati anagrafici dei partecipanti all'esperimento;
- Dati riguardanti il livello di presenza dei soggetti durante l'esperimento;
- Traiettorie compiute dai soggetti;
- Parametri fisiologici dei soggetti.

Questi dati sono stati considerati tutti ad eccezione di quelli riguardanti il livello di presenza dei soggetti.

6.1.1 Descrizione dell'ambiente virtuale

Per la nostra analisi sono risultati particolarmente utili le informazioni relative all'ambiente virtuale nel quale i partecipanti all'esperimento hanno navigato.

Lo studio condotto dagli esperti di realtà virtuale ha reso necessaria la creazione di quattro scenari diversi, variando la qualità grafica degli avatars e il numero di textures utilizzate per ricoprire i vari elementi della scena.

Una descrizione dettagliata dell'ambiente è fornita nel capitolo 5.

Tale descrizione porta ad individuare per ciascuno dei quaranta partecipanti, due variabili:

- *Texture*: utilizzata per fornire informazioni sulla qualità grafica degli edifici virtuali; è definita in termini di numero di textures utilizzate per conferire il materiale alle diverse strutture. E' una variabile binaria che assume due valori, 1 o 2. Il valore 1 indica che l'ambiente nel quale il partecipante ha navigato era caratterizzato da una qualità delle textures bassa (sono state utilizzate venti textures diverse); il valore 2, al contrario, indica una qualità delle textures alta (sono state utilizzate quaranta textures diverse).

- *Avatar*: utilizzata per fornire l'informazione relativa alla qualità grafica degli avatars che popolano il centro commerciale virtuale. E' una variabile binaria che ha due valori possibili, 1 o 2. Il valore 1 indica che l'ambiente nel quale il partecipante ha navigato è popolato da avatar poco realistici (*cartoon form characters*); il valore 2, al contrario, indica la presenza nell'ambiente virtuale di avatar realistici (*Higher fidelity characters*).

6.1.2 Dati anagrafici

La nostra analisi ha richiesto la considerazione dei dati anagrafici e di altre informazioni riguardanti coloro che hanno partecipato all'esperimento. Queste informazioni sono state raccolte dagli esperti di realtà virtuale attraverso un questionario a cui i partecipanti hanno risposto prima della navigazione.

Per ciascun soggetto sottoposto all'esperimento sono state acquisite le seguenti informazioni:

- *Age*: età del soggetto. Per tener traccia di questa informazione è stata utilizzata una variabile numerica.
- *Gender*: sesso del soggetto. E' una variabile binaria che ha due valori possibili, 1 o 2. Il valore 1 indica che il partecipante è di sesso maschile; il valore 2, al contrario, indica che il soggetto è di sesso femminile.
- *Occupy*: attività lavorativa del soggetto. Per sfruttare questa informazione si utilizza una variabile nominale, ovvero una variabile che può assumere un numero limitato di valori; nel caso specifico assume un intero compreso nell'intervallo (1,8). A ciascun intero corrisponde un'attività lavorativa:
 - 1: Undergraduate student;
 - 2: Masters student;
 - 3: PhD student;
 - 4: Research Assistant / Fellow;
 - 5: Staff-systems / technical;
 - 6: Faculty;

- 7: Administrative Staff;
- 8: Other.
- *Medicine*: indica se il soggetto assume abitualmente medicinali. Questa informazione viene fornita attraverso una variabile binaria che ha due valori possibili, 1 o 2. Il valore 1 indica che il partecipante assume regolarmente medicinali; il valore 2, al contrario, indica che il soggetto non assume medicinali.
- *Language*: indica il grado di conoscenza dell'inglese da parte del soggetto. Questa informazione viene indicata con una variabile nominale che ha 3 valori possibili:
 - 1: *basic*;
 - 2: *proficient*;
 - 3: *fluent*.
- *Alcohol*: indica se il soggetto ha assunto sostanze alcoliche nelle sei ore precedenti all'esperimento. Per utilizzare questa informazione si utilizza una variabile binaria che ha due valori possibili, 1 o 2. Il valore 1 indica che il soggetto non ha assunto sostanze alcoliche nel periodo indicato; al contrario il valore 2 indica che il soggetto ha fatto uso di sostanze alcoliche prima dell'esperimento.
- *Literate*: indica il livello di conoscenza informatico. Indicato con un valore compreso tra 1 e 7. Valori vicini ad 1 indicano una conoscenza limitata in campo informatico; valori prossimi a 7 indicano invece una buona conoscenza.
- *Programming*: indica la capacità del soggetto nella programmazione. Indicato con un valore compreso tra 1 e 5. Valori prossimi ad 1 indicano poche conoscenze di programmazione; valori prossimi a 5 indicano un buon livello di esperienza nella programmazione.
- *Virtual reality*: indica il numero di esperienze di realtà virtuale a cui il soggetto ha partecipato.

- **Games:** indica il numero di volte in cui il soggetto ha giocato con un videogame nell'ultimo anno. Per indicare questa informazione si utilizza una variabile nominale che ha un valore compreso tra 1 e 7:
 - 1: never;
 - 2: 1 – 5;
 - 3: 6 – 10;
 - 4: 11 - 15;
 - 5: 16 - 20;
 - 6: 21 – 25;
 - 7: > 25.
- **Gametimes:** indica il numero di ore trascorse dal soggetto a giocare con un videogame in media in una settimana. Si utilizza una variabile nominale che assume un valore compreso tra 1 e 7:
 - 1: 0;
 - 2: < 1;
 - 3: 1 – 3;
 - 4: 3 - 5;
 - 5: 5 – 7;
 - 6: 7 – 9;
 - 7: > 9.

6.1.3 Parametri fisiologici

Durante l'esperimento di realtà virtuale, ciascun partecipante era equipaggiato con speciali dispositivi che consentivano di rilevarne i parametri fisiologici. Tra le informazioni estratte con l'ausilio di questi dispositivi, per il nostro studio sono risultati particolarmente interessanti i dati relativi all'attività cardiaca, alla sudorazione e alla respirazione. La sudorazione e la respirazione sono state rilevate ogni trentaduesimo di secondo a partire dalla fase iniziale di preparazione alla navigazione fino al termine dell'esperimento. L'analisi

dell'attività cardiaca di ciascun soggetto ha consentito di estrarre un'informazione particolarmente interessante: l'*heart rate* ossia il numero di contrazioni al minuto.

Le informazioni relative ai parametri fisiologici sono state utilizzate in vari modi.

6.1.4 Traiettorie

Un'informazione che si è rivelata particolarmente utile è rappresentata dalla traiettoria compiuta dai soggetti durante l'esperimento. La stanza sulle pareti della quale è stato proiettato l'ambiente virtuale, può essere visto come un piano bidimensionale: ad un certo istante i un soggetto si trova nella posizione (x_i, z_i) .

Coloro che hanno condotto l'esperimento hanno tenuto traccia dell'informazione spaziale associata al movimento dei partecipanti: ogni trentaduesimo di secondo oltre a misurare i parametri fisiologici, è stata rilevata la posizione di ciascun partecipante. A partire da queste informazioni spaziali, abbiamo ricostruito l'intera traiettoria compiuta da ciascun soggetto e da questa abbiamo estratto la sottotraiettoria compiuta durante la navigazione vera e propria. L'informazione relativa alla traiettoria è stata utilizzata nella nostra analisi in vari modi come descritto nel paragrafo 6.2 e successivi.

6.1.5 Presence

Oltre ai dati anagrafici, ai valori fisiologici e alle traiettorie compiute dai partecipanti, gli esperti di realtà virtuale ci hanno fornito informazioni sul livello di presenza di ciascun soggetto. Come già spiegato nel capitolo 5 in cui è stato descritto l'esperimento, al termine dell'esperienza virtuale, i partecipanti hanno risposto ad una serie di domande necessarie agli esperti per valutare il grado di presenza e coinvolgimento durante la navigazione. A questo livello dell'analisi si è ritenuto opportuno trascurare queste informazioni perché non abbastanza utili per il tipo di studio che si è deciso di condurre. L'unica

informazione che abbiamo utilizzato relativa alla presenza è il numero di breaks in presence riportati dai soggetti durante l'esperimento. Questa informazione non è stata scartata perché ai fini di uno studio di complementarità tra le discipline di realtà virtuale e data mining, è interessante individuare tecniche per la previsione del numero di BIPs sulla base degli altri dati.

6.2 Analisi dei dati: Fasi e tecniche

La verifica sull'applicabilità del data mining alla realtà virtuale può essere ricondotta all'analisi, con specifiche tecniche di estrazione della conoscenza, di dati estratti durante un esperimento specifico. La scelta delle tecniche di data mining idonee per una tale applicazione richiede la risoluzione di una serie di problemi.

Il primo problema da affrontare riguarda l'obiettivo finale ovvero quali sono le informazioni che si vogliono estrarre. L'obiettivo specifico dipende dal tipo dei dati a disposizione. Ogni esperimento di realtà virtuale viene realizzato per uno specifico scopo: l'ambiente ricreato e i dati estratti variano da situazione a situazione.

Nel nostro caso, presi in considerazione i dati estratti dall'esperimento di realtà virtuale, potrebbe essere interessante trovare delle affinità (in termini di valore degli attributi) tra i soggetti coinvolti.

Di particolare interesse è l'analisi delle traiettorie compiute dai partecipanti: potrebbe esistere un legame tra la traiettoria, intesa come percorso compiuto da un soggetto e i suoi parametri fisiologici o/e i suoi dati anagrafici. Sarebbe interessante scoprire cose del tipo *“I soggetti più esplorativi, la cui traiettoria attraversa l'intero ambiente virtuale, hanno età compresa tra 23 e 28 anni”* o

ancora *“Se un soggetto ha un numero di BIPs inferiore a 3 allora la sua traiettoria è lineare”*.

Dal punto di vista della realtà virtuale sarebbe interessante trovare delle correlazioni tra il numero di BIPs e le altre informazioni relative all'esperimento. Il problema della presenza in un ambiente virtuale è un argomento di studio per gli esperti del settore e scoprire quali siano i fattori che influenzano la segnalazione dei BIPs può risultare di notevole interesse. Potrebbe essere utile in questo caso scoprire fatti del tipo *“I soggetti di età inferiore ai 20 anni, con una variazione della sudorazione pari a $\pm 1\%$, hanno un numero di BIPs compreso tra 4 e 5”*.

Poiché l'obiettivo di questo lavoro è quello di soddisfare sia le esigenze informative del data mining sia quelle di realtà virtuale, l'analisi è stata condotta in modo da garantire dei risultati interessanti per entrambi gli ambiti di studio.

L'approccio utilizzato per l'analisi dei dati prevede due fasi fondamentali:

- 1) studio delle traiettorie compiute dai soggetti e successiva suddivisione di questi ultimi in gruppi, in modo che soggetti appartenenti allo stesso gruppo compiano una traiettoria simile;
- 2) Questa seconda fase si articola a sua volta in due fasi secondo l'obiettivo specifico dell'analisi:
 - 2a) Studio delle affinità (in termini dei valori dei vari attributi) tra i soggetti appartenenti allo stesso cluster (definito in 1)). Successiva definizione di regole per la predizione del cluster di appartenenza di un soggetto in base al valore degli altri attributi;
 - 2b) Studio delle affinità tra soggetti caratterizzati dallo stesso numero di BIPs. Successiva definizione di regole che consentano di predire il numero di BIPs in base ai valori degli altri attributi. In questo caso i gruppi determinati come nel punto 1) vengono utilizzati come un

semplice attributo che contribuisce a caratterizzare le regole per la predizione del numero di BIPs.

Definita la strategia per analizzare i dati, è necessario individuare le tecniche di data mining per realizzare le varie fasi del processo che è stato definito. Per quanto riguarda la prima fase, in cui l'obiettivo è raggruppare insieme traiettorie simili, la tecnica che si presta meglio ad essere utilizzata è il clustering. Come già spiegato dettagliatamente nel capitolo 3, il clustering consente la suddivisione dei dati in gruppi o clusters; il raggruppamento è realizzato in modo che tra gli oggetti appartenenti allo stesso clusters ci sia un'alta similarità, mentre tra oggetti appartenenti a clusters diversi ci sia una bassa similarità. La similarità tra le traiettorie può essere calcolata in vari modi: l'algoritmo di clustering utilizzato valuta la similarità in termini di distanza euclidea. Una descrizione dettagliata dell'algoritmo utilizzato è fornita nel capitolo 3.

La seconda fase dell'analisi, in cui si vuole definire un attributo (cluster di traiettorie o numero di BIPs) in termini degli altri, richiede il ricorso alla classificazione. Questa tecnica consente di definire delle regole con le quali si può predire il valore di un attributo (target o classe) in termini degli altri attributi. Le regole così estratte vengono utilizzate per predire la classe di un particolare oggetto nel caso in cui di esso si conoscano i valori di tutti gli attributi a eccezione della classe di appartenenza. Una descrizione dettagliata su come viene realizzata la classificazione e sull'uso delle regole di decisione prodotte, è fornita nel capitolo 3.

6.3 Studio delle traiettorie: ricostruzione

La prima fase dell'analisi prevede la suddivisione dei soggetti sottoposti all'esperimento in gruppi in modo che, soggetti appartenenti allo stesso gruppo siano molto simili e risultino altrettanto dissimili da qualunque soggetto appartenente ad altri gruppi. La similarità o la dissimilarità tra due oggetti viene valutata in base ad uno o più attributi. Nel caso specifico della nostra analisi questa valutazione è stata fatta considerando la traiettoria compiuta da ciascun soggetto: risulteranno nello stesso gruppo i soggetti che hanno una traiettoria molto simile, in termini di coordinate spaziali, lunghezza e andamento. Questo tipo di raggruppamento ha richiesto l'utilizzo di un algoritmo di clustering specifico per il trattamento di traiettorie. Tale algoritmo richiede come input il file contenente le informazioni sulle traiettorie compiute da ciascun soggetto. In questo caso per traiettoria s'intende una sequenza di n punti $(x_1, z_1), (x_2, z_2), \dots, (x_n, z_n)$ definiti su uno spazio bidimensionale.

Il problema sorto in questo contesto è relativo all'organizzazione dei dati nel file di input.

6.3.1 Ricostruzione: Aspetti generali

I dati riguardanti le traiettorie di coloro che hanno partecipato all'esperimento di realtà virtuale sono stati organizzati in due files diversi dagli esperti che l'hanno condotto: in uno sono state memorizzate le posizioni di ciascun soggetto lungo l'asse x , nell'altro le posizioni lungo l'asse z . Una tale strutturazione dei dati non si prestava al tipo di analisi scelto; l'algoritmo di clustering richiedeva i dati organizzati in un certo modo, diverso da quello in cui sono stati forniti. Per la risoluzione di questo problema sono state valutate due alternative:

- 1) Modifica dell'algoritmo di clustering in modo che potesse accettare i dati organizzati come previsto dagli esperti di realtà virtuale;

- 2) Riorganizzazione in un unico file dei dati riguardanti le traiettorie. Un tale approccio non richiede una modifica dell'algoritmo di clustering se i dati all'interno del nuovo file sono organizzati nel formato richiesto dall'algoritmo.

La nostra scelta è stata quella di riorganizzare i dati in modo da non rendere necessaria una modifica dell'algoritmo di clustering. Seguendo la prima alternativa, il rischio sarebbe stato quello di ottenere un codice complesso e poco leggibile senza che se ne potesse trarre un vantaggio in termini di complessità dell'algoritmo. Il secondo approccio invece, oltre a facilitare la fase di clustering, ci ha consentito di darne una rappresentazione grafica con MatLab 7.0 senza alcuna difficoltà. Nei prossimi paragrafi è fornita una descrizione più dettagliata dei files contenenti i dati sulle traiettorie nella loro forma originaria, del file nel formato richiesto dall'algoritmo di clustering e dell'algoritmo per la riorganizzazione dei dati.

6.3.2 Organizzazione dei dati di input

Il processo di rilevamento della posizione dei vari soggetti durante l'esperimento ha prodotto due file contenenti rispettivamente le coordinate spaziali lungo l'asse x e lungo l'asse z . Per ciascun soggetto le misurazioni sono state fatte a intervalli di un trentaduesimo di secondo. Considerando che il rilevamento è iniziato prima della navigazione vera e propria ed è terminato nella fase post-navigazione, il numero di posizioni rilevate per ciascun soggetto sono dell'ordine delle migliaia.

I dati in ciascuno di questi files sono organizzati per colonne: ci sono tante colonne quanti sono i soggetti di cui è stata rilevata la posizione. Nel caso specifico, ciascun file ha i dati organizzati su 40 colonne: la colonna i contiene le coordinate lungo l'asse x (resp. z) del soggetto i . La prima riga in ciascuna colonna contiene un valore che identifica univocamente il soggetto i i cui dati sono memorizzati in quella colonna. Tornando all'esempio, se la prima riga

della colonna i contiene il valore 12111 allora i valori memorizzati in quella colonna sono relativi al soggetto 12111. In figura 6.1 è riportata una descrizione dell'organizzazione dei files che ci sono stati forniti dagli esperti di realtà virtuale.

Id_{e1}	Id_{e2}	Id_{e_m}
.....
$X_1^1 (Z_1^1)$	$X_2^1 (Z_2^1)$	$X_m^1 (Z_m^1)$
$X_1^2 (Z_1^2)$	$X_2^2 (Z_2^2)$	$X_m^2 (Z_m^2)$
$X_1^3 (Z_1^3)$	$X_2^3 (Z_2^3)$	$X_m^3 (Z_m^3)$
.....
$X_1^n (Z_1^n)$	$X_2^n (Z_2^n)$	$X_m^n (Z_m^n)$

Figura 6.1 Organizzazione del file contenente le coordinate spaziali lungo l'asse x (rispett. asse z)

L'algoritmo di clustering utilizzato richiede un'organizzazione diversa dei dati: le coordinate spaziali devono essere memorizzate in unico file e secondo uno specifico schema. Tale algoritmo richiede che il file delle traiettorie abbia i dati organizzati su quattro colonne. Ogni riga contiene quindi quattro informazioni: l'identificativo della traiettoria, l'istante di tempo in cui è stata rilevata la posizione, la coordinata lungo l'asse x e la coordinata lungo l'asse z. Tra i records del file è definito un ordinamento: il primo attributo rispetto al quale si ordinano i dati è l'identificatore. Per ciascuna traiettoria un secondo ordinamento viene fatto in base al tempo. Quindi per ciascun soggetto, le coordinate sono memorizzate nell'ordine in cui sono state rilevate. In figura 6.2 è riportato un esempio in cui si mostra come devono essere organizzati i dati nel file fornito in input all'algoritmo di clustering.

Ide₁	t₁¹	X₁¹	Z₁¹
Ide₁	t₁²	X₁²	Z₁²
.....
Ide₁	t₁ⁿ	X₁ⁿ	Z₁ⁿ
Ide₂	t₂¹	X₂¹	Z₂¹
.....
Ide_m	t_m¹	X_m¹	Z_m¹
Ide_m
Ide_m	t_m^k	X_m^k	Z_m^k

Figura 6.2 Organizzazione dei dati nel file di testo fornito in input all'algoritmo di clustering

In appendice A sono riportati i frammenti di alcuni file di dati per dare un'idea più chiara dell'organizzazione.

6.3.3 Ricostruzione delle traiettorie: algoritmo

La creazione del file delle traiettorie nel formato richiesto dall'algoritmo di clustering ha richiesto l'implementazione di un algoritmo. Una tale soluzione si è resa necessaria per le notevoli dimensioni dei files originali e per l'impossibilità quindi di elaborarli manualmente. L'implementazione dell'algoritmo è stata fatta nel linguaggio C. Tra tanti linguaggi possibili, vista l'assenza di vincoli su quale adottare, si è scelto il C per la semplicità dei costrutti sintattici che offre per la gestione dei files.

La procedura per la riorganizzazione delle coordinate spaziali si articola in diverse fasi:

1. Si accede al file *timing* che, per ciascun partecipante, tiene traccia degli istanti di tempo in cui iniziano le varie fase dell'esperimento. Per ciascun partecipante si individuano gli istanti di tempo in cui inizia e termina la fase di navigazione. La fine della navigazione coincide con l'inizio della fase successiva denominata *dark sky event*. Indichiamo convenzionalmente l'istante di tempo in cui il soggetto *i* inizia l'esperimento vero e proprio con

$start(i)$; indichiamo con $end(i)$ l'istante in cui termina la navigazione e inizia la fase successiva.

In appendice A è fornita una descrizione dell'organizzazione dei dati nel file *timing*.

2. Poiché il nuovo file che deve essere restituito in output deve contenere per ciascuna coppia di coordinate, l'istante di tempo in cui viene rilevata, si tiene traccia del tempo. Indichiamo la variabile utilizzata per questo scopo con *time*; inizialmente *time* ha valore 0. Per ogni rilevazione tale variabile viene incrementata di un valore pari ad un trentaduesimo. La posizione nello spazio viene infatti rilevata ogni trentaduesimo di secondo.
3. Si crea un nuovo file in cui, per ciascun partecipante, si memorizzano identificatore, *time*, coordinata lungo l'asse x e coordinata lungo l'asse z. Tale operazione viene fatta per il partecipante *i* se e solo se $start(i) \leq time < end(i)$.
4. Ordinamento del file per identificatore e *time*.

L'output di questo algoritmo può essere utilizzato per ottenere una rappresentazione grafica delle traiettorie.

Terminato il processo di creazione del file delle traiettorie è possibile eseguire il clustering.

In figura 6.3 è fornita una rappresentazione delle traiettorie ottenuta con MatLab 7.0. Nella rappresentazione è riportata anche la struttura dell'ambiente ricreato delineato in nero. Come risulta dalla figura, il comportamento di ciascun soggetto è diverso. Alcuni soggetti visitano l'intero ambiente virtuale, altri si limitano a seguire un percorso lineare. Alcuni soggetti seguono un percorso che non rispetta la struttura del mondo artificiale, attraversando ad esempio i muri virtuali.

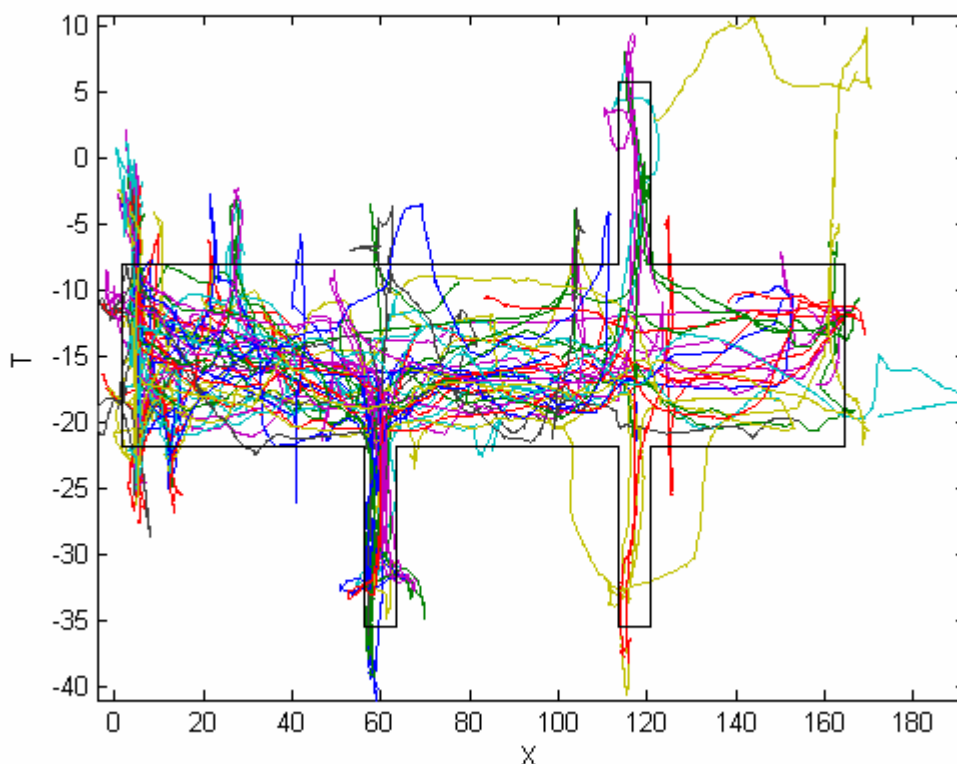


Figura 6.3 Rappresentazione delle traiettorie nel centro commerciale virtuale

6.4 Clustering

La prima fase dell'analisi dei dati di realtà virtuale prevede l'individuazione di gruppi di traiettorie che risultino simili. La tecnica che si presta meglio a risolvere un problema di questo tipo è il clustering. L'applicazione di un algoritmo di clustering su un insieme di traiettorie consente l'individuazione di gruppi di soggetti tali che i soggetti appartenenti allo stesso cluster compiono una traiettoria molto simile. Inoltre la traiettoria compiuta da un soggetto risulta essere molto diversa dalla traiettoria compiuta da qualunque altro soggetto appartenente ad un gruppo diverso.

Esistono vari algoritmi per realizzare il clustering di traiettorie, ciascuno di essi offre vantaggi in situazioni diverse ed a seconda del tipo di dato fornito in input. In generale il clustering risultante dall'utilizzo di questi algoritmi è tanto migliore quanto maggiore è il numero di traiettorie.

Nel nostro caso, viste le caratteristiche delle traiettorie di realtà virtuale, non esiste nessuna garanzia sulla qualità del clustering. Le traiettorie oggetto del nostro studio sono molto particolari: il numero è esiguo (quaranta) ed inoltre ciascuna di esse è definita con un numero elevato di punti (fino ad un massimo di ventiquattromila punti). Gli algoritmi di clustering esistenti in genere vengono applicati su un numero maggiore di traiettorie. Con un numero basso di traiettorie, come nel nostro caso, non è garantito un risultato significativo.

Vista la particolarità delle traiettorie di realtà virtuale, si è scelto di considerare contemporaneamente due algoritmi di clustering, uno di tipo k-means e uno di tipo gerarchico. Questi due algoritmi di clustering sono stati descritti in maniera approfondita nel capitolo 3. L'aver scelto due diversi algoritmi di clustering ha come conseguenza che anche la fase successiva di classificazione si ramifica in due analisi separate: una in cui si considerano i clusters determinati con il k-means e una in cui si considerano i clusters determinati con l'algoritmo gerarchico.

Entrambi gli algoritmi richiedono che l'utente fornisca il numero k di clusters che si vogliono ottenere. La scelta di k non è banale e dipende dal numero di oggetti su cui realizzare il clustering. Nel nostro caso, avendo solo quaranta traiettorie, non avrebbe senso scegliere un k maggiore di venti; una tale scelta porterebbe ad ottenere clusters che al massimo contengono due oggetti. Se così fosse non avrebbe senso la fase successiva di classificazione, in quanto minore è il numero di oggetti all'interno di un cluster e più bassa è la probabilità di trovare tra di essi delle affinità. Nel caso in cui ad un cluster appartenga un solo oggetto, la descrizione scaturirebbe banalmente dagli attributi associati all'oggetto stesso, senza che sia necessaria la fase di

classificazione. Queste considerazioni portano ad escludere alcune scelte sul k iniziale. Individuati i k significativi, si esegue l'algoritmo di clustering con input il file dei dati e l'intero k : l'algoritmo viene eseguito tante volte quanti sono i k significativi.

Terminata questa fase, si procede nella scelta del k per il quale l'algoritmo offre un buon risultato. La nostra scelta, nel caso del clustering realizzato con il k -means è stata di $k= 5$ e $k= 6$. Questa scelta è scaturita dalla constatazione che, per questi valori, i clusters si stabilizzano; questo significa che passando da un numero di clusters pari a 5 ad uno pari a 6, i clusters risultano in linea di massima gli stessi.

Anche nel caso di clusters determinati con l'algoritmo gerarchico la scelta si è concentrata sulla suddivisione delle traiettorie in cinque e sei clusters. I motivi di tale scelta sono due: da un lato si è constatato che anche in questo caso, l'algoritmo si stabilizzava con questi valori del k ; dall'altro è necessario un certo livello di consistenza tra l'analisi condotta considerando i clusters determinati con il k -means e quella condotta considerando i clusters determinati con l'algoritmo gerarchico.

Da un confronto tra i clusters determinati con parametro $k= 5$ e $k= 6$ con i due diversi algoritmi si è constatato che i gruppi di traiettorie sono in linea di massima gli stessi. I clusters coincidono con una percentuale di circa il 70%.

In appendice B è riportata una tabella dove per ciascuna traiettoria è indicato il cluster di appartenenza, a seconda dell'algoritmo di clustering utilizzato e del numero di clusters k . E' riportata inoltre una rappresentazione grafica di ciascun cluster.

6.5 Clustering con k-means: traiettorie rappresentative

L'output fornito dall'algoritmo di clustering applicato ad un insieme di traiettorie è un insieme di clusters; ciascun cluster è costituito da una o più traiettorie ed una traiettoria appartiene ad uno ed un solo cluster. In generale, le traiettorie che appartengono ad uno stesso cluster risultano essere molto simili. La similarità in questo contesto è da intendersi definita in termini di distanza euclidea. Può essere interessante capire qual è l'andamento medio all'interno di ciascun cluster; questo equivale a calcolare la traiettoria media di ciascuno di essi. Ci siamo limitati a calcolare la traiettoria media dei clusters determinati con il k-means, sia per $k=5$ che per $k=6$.

6.5.1 Traiettorie rappresentative: aspetti generali

Dato un insieme di traiettorie suddivise in k clusters, la definizione della traiettoria media di ciascun gruppo richiede la considerazione delle coordinate di tutte le traiettorie che vi appartengono. La traiettoria media di un cluster, come una qualsiasi traiettoria, è definita come una sequenza di punti in uno spazio: ciascuno di questi punti è definito come la media dei corrispondenti punti delle traiettorie che appartengono al cluster. La corrispondenza è definita in termini temporali: un punto della traiettoria media del cluster m rilevato all'istante t , p_m^t , è calcolato come valor medio dei punti delle traiettorie appartenenti al cluster m e rilevati all'istante temporale t .

Un'operazione di questo tipo è possibile se:

- le traiettorie appartenenti allo stesso cluster hanno la stessa lunghezza, sono definite cioè dallo stesso numero di punti;
- le traiettorie sono allineate temporalmente, i soggetti iniziano la navigazione nello stesso istante di tempo e (in base al punto precedente) terminano nello stesso istante.

6.5.2 Riallineamento temporale delle traiettorie

Le traiettorie oggetto del nostro studio non hanno la stessa lunghezza e di conseguenza non sono allineate temporalmente.

Questo è conseguenza del fatto che la traiettoria considerata per la nostra analisi è quella compiuta durante la navigazione vera e propria; tale traiettoria rappresenta solo una porzione dell'intero percorso compiuto dal partecipante a partire dalle fase preliminari di preparazione fino alla fase post-esperimento.

Le “sotto-traiettorie” considerate non sono allineate temporalmente in quanto la fase preliminare all'esperimento non ha la stessa durata per tutti i partecipanti; può accadere, per esempio, che un soggetto abbia più difficoltà di altri a prendere dimestichezza con gli strumenti di navigazione e quindi richieda più tempo per capirne il modo d'uso. Una situazione di questo tipo comporta che i soggetti inizino la navigazione in istanti temporali diversi. A questo problema si aggiunge il fatto che la lunghezza delle traiettorie non sia sempre la stessa ma vari di uno o due punti. Una soluzione presa in considerazione per il calcolo delle traiettorie medie è quello di allineare temporalmente le traiettorie ripetendo i punti estremi delle traiettorie più brevi in maniera tale da avere traiettorie tutte della stessa lunghezza. Un approccio di questo tipo si basa sull'idea che è da considerarsi traiettoria anche quella costituita da un solo punto. Una tale situazione si verifica se un soggetto rimane fermo in una posizione per un certo intervallo di tempo.

Questa soluzione è stata scartata perché il fine ultimo del calcolo delle traiettorie medie è quello di darne una rappresentazione per capire quale sia l'andamento all'interno di ciascun cluster.

La soluzione adottata prevede quindi il calcolo di ogni punto della traiettoria ad un certo istante di tempo come la media dei punti delle traiettorie rilevate a quell'istante di tempo. Una traiettoria contribuisce al calcolo di un determinato punto se l'istante di tempo a cui si riferisce rientra nell'intervallo di navigazione del soggetto.

6.5.3 Traiettorie rappresentative: algoritmo

L'algoritmo implementato per individuare le traiettorie rappresentative si articola nei seguenti passi:

- 1) Per ciascun soggetto i si memorizza l'istante di tempo in cui inizia la navigazione e quello in cui termina. Indichiamo queste informazioni rispettivamente con $start(i)$ ed $end(i)$. Questi dati sono memorizzati nel file *timing* che per ciascun soggetto riporta gli istanti di tempo in cui hanno inizio le varie fasi dell'esperimento. Ciò che si indica come $start(i)$ corrisponde a *start of the actual experiment*; $end(i)$ corrisponde a *dark ski event* (la navigazione ha termine quando inizia la fase successiva).
- 2) Per ciascun cluster j si determina il primo istante di tempo in cui uno dei soggetti che vi appartiene inizia la fase di navigazione. Questa informazione viene acquisita cercando il minimo tra gli $start(i)$ delle traiettorie $i \in cluster(j)$. Indichiamo tale valore con min_j . In maniera analoga si calcola l'istante di tempo massimo in cui esiste ancora una qualche traiettoria appartenente al cluster j per cui non è terminata la fase di navigazione. Questa informazione viene acquisita cercando il massimo tra gli $end(i)$ delle traiettorie $i \in cluster(j)$. Indichiamo tale istante di tempo con max_j .
- 3) Per ciascun cluster j si calcolano i punti della traiettoria media. Partendo dall'istante di tempo $time = min_j$ si calcola il primo punto della traiettoria rappresentativa. Il punto p di una traiettoria i appartenente al cluster j contribuisce al calcolo se:
 - il punto p è stato rilevato all'istante $time$;
 - $start(i) \leq time < end(i)$, ovvero il soggetto che ha compiuto la traiettoria i all'istante $time$ è in fase di navigazione.

Si incrementa $time$ di un trentaduesimo e si calcola il punto successivo della traiettoria media del cluster j . Il calcolo dei punti della traiettoria viene

ripetuto fino a quando non è verificata la condizione di terminazione $time > max_j$.

6.5.4 Rappresentazione grafica delle traiettorie medie.

Nelle figure seguenti sono riportate le traiettorie medie nel caso di clusters determinati con il k-means con $k=5$ e $k=6$. Come risulta da tali rappresentazioni il comportamento all'interno di ciascun cluster non è ben definito. Alcuni clusters sono caratterizzati da un andamento medio molto simile, ad esempio i clusters 0 e 3 determinati con $k=5$. In situazioni come questa appena illustrata, data una traiettoria i che ha un certo andamento, la determinazione del cluster di appartenenza può risultare complicata se ci si basa solo sull'andamento medio all'interno di ciascun gruppo.

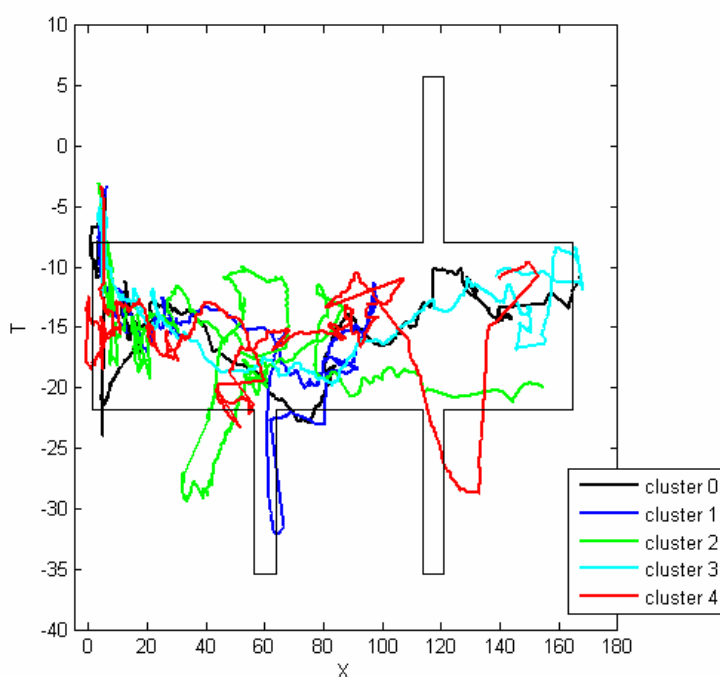


Figura 6.4 Rappresentazione delle traiettorie medie nel caso $k=5$

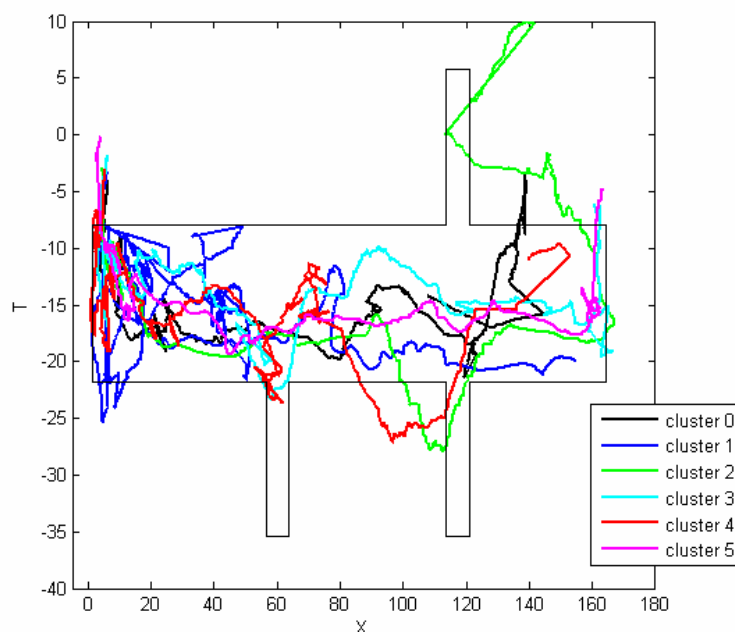


Figura 6.5 Rappresentazione delle traiettorie medie nel caso $k=6$

Fornire una descrizione di ciascun cluster considerando esclusivamente la traiettoria media può risultare particolarmente difficile. Un approccio di questo tipo offre buoni risultati quando il numero di traiettorie è molto alto. In questo caso infatti i clusters determinati risulterebbero più compatti, in termini di occupazione spaziale e di conseguenza l'andamento medio all'interno di ciascun cluster sarebbe distinto da quello degli altri.

Nel nostro caso, l'individuazione delle traiettorie rappresentative non è sufficiente a causa del numero esiguo di traiettorie. E' necessario quindi un affinamento dell'analisi utilizzando a tale scopo tecniche di classificazione.

6.6 Clusters: Esempi

Nelle figure 6.6 e 6.7 sono riportate rispettivamente le traiettorie appartenenti al cluster 0 e quelle appartenenti al cluster 4, nel caso di clusters determinati con il k-means con $k=5$.

Il cluster 0 è caratterizzato da traiettorie che visitano l'intero ambiente virtuale, potrebbe essere definito come il cluster degli “esploratori”.

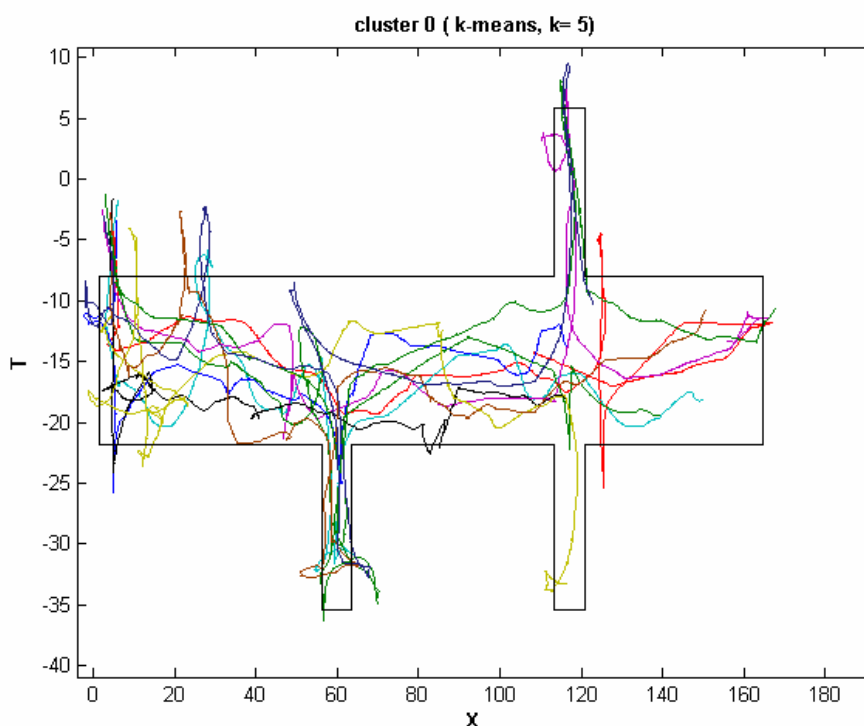


Figura 6.6 Rappresentazione delle traiettorie appartenenti al cluster 0

Il cluster 3 è caratterizzato da soggetti che compiono una traiettoria lineare, se si esclude qualche eccezione. Da una tale rappresentazione può risultare che siano un po' meno esperti in navigazione virtuale rispetto a quelli appartenenti al cluster 0, non rispettando completamente i limiti dell'ambiente.

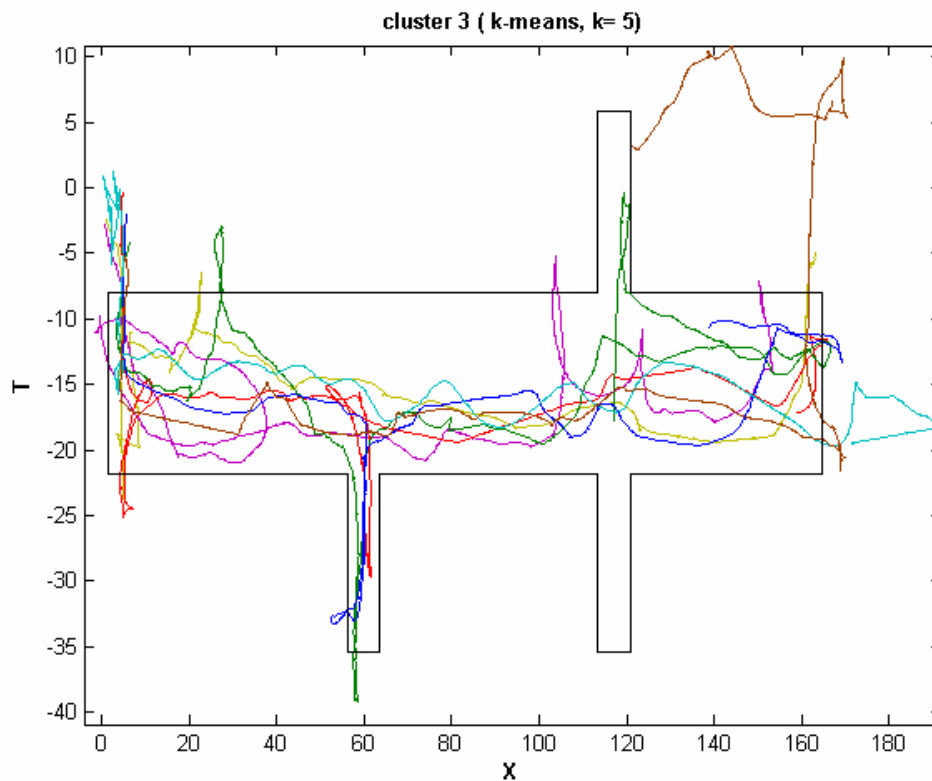


Figura 6.7 Rappresentazione delle traiettorie appartenenti al cluster 3

6.7 Classificazione

Il processo di clustering applicato alle traiettorie di realtà virtuale, oltre ad aver consentito l'individuazione di gruppi di traiettorie simili, ha permesso la definizione di un nuovo attributo da associare a ciascun soggetto. Il cluster di appartenenza di una certa traiettoria diventa un nuovo dato associato al soggetto che la compie così come lo sono età, occupazione, sesso. Non essendo sufficiente dare una descrizione di ciascun cluster in termini di andamento medio al suo interno, può essere utile cercare delle affinità tra le traiettorie che vi appartengono, espresse in funzione degli altri attributi. La tecnica che meglio si presta per realizzare questo tipo di studio è la

classificazione. Questa tecnica per poter essere applicata richiede in input un insieme di oggetti e l'indicazione dell'attributo target.

6.7.1 Descrizione del dataset

Nella nostra analisi, il dataset di oggetti è rappresentato dalla descrizione dei soggetti in termini di dati anagrafici, parametri fisiologici, traiettoria compiuta (o meglio cluster di appartenenza). Per quanto riguarda la variabile target sono state fatte varie classificazioni con classi diverse. L'algoritmo di classificazione, oltre a richiedere le informazioni sui dati da analizzare, richiede l'indicazione di altre informazioni. Questi parametri aggiuntivi forniscono indicazioni sul tipo di classificazione da effettuare. Uno dei parametri più interessanti è quello con il quale si indica all'algoritmo la necessità di effettuare una fase di pruning sui dati. Come già spiegato dettagliatamente nel capitolo 3, l'albero di decisione prodotto da un algoritmo di classificazione ha associato un certo livello di accuratezza. L'output dell'algoritmo di classificazione non è valido in assoluto ma ha associato una certa percentuale di errore. Una regola del tipo *"Se un soggetto ha 20 anni allora appartiene al cluster 3"* può non essere valida in assoluto. Non è detto che tutti i partecipanti di vent'anni appartengano al cluster 3. Se otto dei quaranta soggetti hanno vent'anni ma solo cinque di loro appartengono al cluster, allora la regola è valida nel 60% dei casi. Nel calcolo della confidenza delle regole estratte si può tener conto solo delle tuple così come sono definite; in alternativa si può tener conto anche di un errore intrinseco ai dati. In quest'ultimo caso si ipotizza che i dati sulla base dei quali si classifica possano non essere completamente corretti (ad esempio possono essere stati introdotti degli errori durante la creazione del datasets). La fase di pruning viene eseguita proprio per tener conto dell'errore intrinseco ai dati e consentire così la modifica dell'albero di decisione prodotto e della confidenza associata alle regole che se ne estraggono. In questa prima fase si è ipotizzato che i dati forniti non fossero completamente corretti; questo ha richiesto

l'indicazione all'algoritmo di classificazione della necessità di una fase di pruning.

6.7.2 Attributo target

Il primo problema da affrontare in fase di classificazione riguarda la scelta dell'attributo classe ovvero dell'attributo, tra quelli che descrivono un insieme di oggetti, di cui si vuole definire un modello. Il modello è rappresentato da un insieme di regole che consentono di predire il valore della classe sulla base dei valori degli altri attributi. La scelta dell'attributo classe dipende dalle esigenze informative di chi effettua la classificazione. Nel nostro caso si hanno due tipi di esigenze: da un lato l'esigenza di poter descrivere le traiettorie appartenenti allo stesso cluster in termini degli altri attributi; dall'altro la necessità di individuare un modello per la previsione dei breaks in presence. Questi problemi hanno richiesto di realizzare più classificazioni, alcune considerando come classe l'attributo *cluster_di_appartenenza* associato a ciascuna traiettoria; altre considerando come classe il numero di BIPs rilevati durante la navigazione per ciascun soggetto.

6.7.3 Trattamento dei parametri fisiologici

Tra le informazioni utilizzate per definire un modello per i clusters di traiettorie o per il numero di BIPs sono state considerati i parametri fisiologici dei partecipanti all'esperienza virtuale e riscontrati durante l'esperimento. Come mostrato in appendice A, le informazioni fisiologiche sono state rilevate a intervalli regolari. Quindi diversamente dai dati anagrafici in cui si ha un solo valore per ciascun soggetto, per i parametri fisiologici si hanno a disposizione tante misurazioni, se si esclude l'heart rate per il quale si ha un solo valore.

Il problema che ci si è posti riguarda l'utilizzo di queste informazioni. In questa fase dell'analisi si è scelto di considerarle allo stesso modo degli altri attributi. Questo ha richiesto il calcolo di un valore per ciascun parametro fisiologico,

che fornisca un'indicazione generale sull'andamento. Si è scelto quindi di associare a ciascun soggetto il valore medio della sua sudorazione, il valore medio della sua respirazione e l'heart rate. La determinazione dei valori medi dei parametri fisiologici di ciascun soggetto, ha richiesto l'elaborazione dei file contenenti le misurazioni e di conseguenza l'implementazione di un algoritmo per il calcolo dei valori rappresentativi per questi dati.

I passi eseguiti dall'algoritmo sono i seguenti:

- 1) per ciascun soggetto i si memorizza l'istante di tempo in cui inizia la navigazione e quello in cui termina. Indichiamo queste informazioni con $start(i)$ ed $end(i)$; ciò che ci interessa sono infatti i parametri fisiologici durante la navigazione;
- 2) Per ciascun soggetto i si calcola la media della sudorazione e della respirazione. Un valore della sudorazione (risp. respirazione) m_i^t rilevato all'istante di tempo t contribuisce al calcolo della media di quel parametro se $start(i) \leq t < end(i)$.

6.7.4 Target cluster: strategia *ClusterNotCluster*

Alle regole di decisione estratte con il processo di classificazione è associata un'informazione molto interessante: la *confidenza*, vale a dire la probabilità che la conclusione della regola sia verificata sapendo che lo è la premessa.

La classificazione realizzata sui dati di realtà virtuale con classe da predire il cluster di appartenenza ha consentito l'individuazione di poche regole caratterizzate da una buona confidenza. Per migliorare la classificazione, nel caso in cui la classe sia il cluster di appartenenza, si è scelto di adottare una nuova strategia per la definizione dell'attributo target. Questo nuovo approccio si basa sul fatto che l'attributo cluster di appartenenza associato ad ogni traiettoria è una variabile che ha cinque o sei valori possibili (a seconda che si consideri $k=5$ o $k=6$). Considerando come classe un attributo di questo tipo, dall'albero di decisione si ricavano delle regole che descrivono i cinque o sei

clusters. La nuova strategia si basa sull'ipotesi che considerando una classe con un numero di valori possibili inferiore, l'accuratezza delle regole estratte migliori. L'approccio prevede di associare a ciascuna traiettoria un nuovo attributo che va a sostituirsi all'attributo *cluster di appartenenza*. Questo attributo è rappresentato da una variabile binaria che assume due valori possibili, n o not_n con n che varia nell'intervallo $[0,4]$ o $[0,5]$. E' prevista l'esecuzione di tante classificazioni quanti sono i clusters di traiettorie per ricavare degli alberi di decisione in cui si predice un unico cluster. Per esempio, si consideri la suddivisione delle traiettorie in 5 clusters. Nell'ipotesi di classificare e di voler definire un modello per il cluster 0, il nuovo attributo avrà valore 0 per quelle traiettorie che appartengono effettivamente al cluster 0; il nuovo attributo avrà valore *not 0* per tutte le altre traiettorie. Questo implica che se si considerano i clusters creati con parametro $k=5$, nelle cinque classificazioni con il nuovo attributo, il valore assunto per questo nuovo attributo da una traiettoria avrà valore n ($n=0,...,4$) solo nella classificazione in cui si vuole definire il modello del cluster a cui appartiene. In tutti gli altri casi avrà valore *not n*.

L'albero di decisione fornito in output da un algoritmo di classificazione, con l'utilizzo di un attributo di questo tipo come classe, è caratterizzato dall'avere foglie che hanno due possibili valori: n o $not n$. Si è pensato che riducendo i valori associati all'attributo classe si potessero ottenere risultati migliori in termini di confidenza. L'ipotesi che ha portato all'utilizzo di questo nuovo approccio è stata in parte dimostrata. Effettivamente questo approccio ha consentito l'individuazione di alberi di decisione decisamente migliori in termini di confidenza; è stato registrato un miglioramento nell'accuratezza associata all'intero albero. Un'analisi attenta delle singole regole che descrivono i vari clusters però, ha messo in luce una confidenza vicina a quella delle regole individuate con l'approccio consueto (considerando l'attributo cluster di appartenenza). Il miglioramento dell'accuratezza generale associata all'albero

è riconducibile alla confidenza associata alle regole per la previsione della classe *not cluster n*. Queste regole però non sono particolarmente interessanti perché lo scopo per il quale è stato utilizzato quest'approccio è quello di caratterizzare meglio gli oggetti etichettati con *cluster n*.

In figura 6.8 è riportato l'albero di classificazione risultante da un processo di classificazione per la predizione del cluster 0. Per questa classificazione sono stati considerati i dati anagrafici dei soggetti, i parametri fisiologici (esclusa la media heart rate) e le caratteristiche dell'ambiente virtuale.

I nodi foglia grigi rappresentano l'etichetta *not 0* mentre quelle blu rappresentano l'etichetta 0.

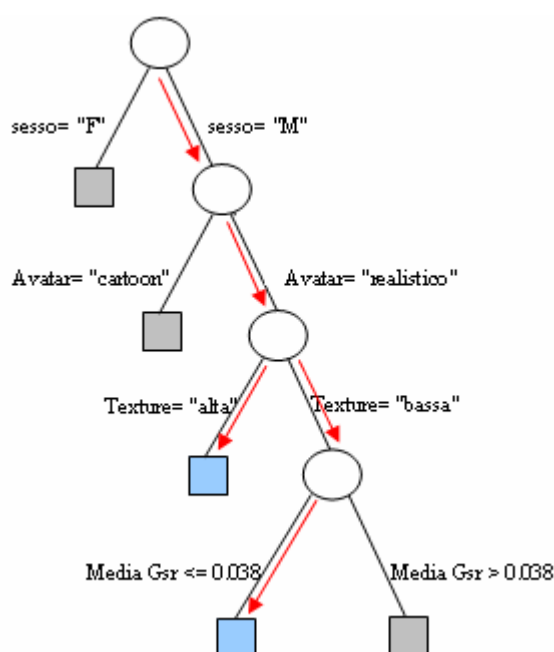


Figura 6.8 Albero di classificazione

La percentuale di errore associata all'albero è 12%, un valore decisamente basso se confrontato con l'accuratezza degli alberi di decisione risultanti con l'utilizzo dell'approccio tradizionale (riportati in appendice B).

Dall'albero riportato in figura 6.8 si estraggono le seguenti regole:

- Se ((Sesso= 'maschio') & (Avatar='realistico') & (Texture= 'alta')) allora numero_cluster= 0 (5.0/2.3)
- Se ((Sesso= 'maschio') & (Avatar='realistico') & (Texture= 'bassa') & (MediaGSR<=0.038)) allora numero_cluster=0 (2.0/1.0)

Il rapporto indicato tra parentesi accanto ad ogni regola ne indica la confidenza. Per esempio, la prima regola deve essere interpretata in questo modo: su cinque soggetti i cui attributi soddisfano la premessa solo 2.7 appartengono al cluster 0. Come risulta da un confronto con le regole riportate in appendice B, la confidenza delle singole regole non ha subito grandi variazioni. Il numero di casi in cui è soddisfatta la premessa, ma non la conclusione, spesso non ha un valore intero. Questo è dovuto alla fase di pruning in cui si fa una stima dell'errore intrinseco ai dati. Nell'ipotesi che i dati forniti in input all'algoritmo di classificazione fossero corretti, non sarebbe necessaria la fase di pruning. In questo caso il valore indicato come denominatore nel rapporto che rappresenta la confidenza sarebbe nettamente inferiore (di conseguenza la confidenza sarebbe più alta).

6.7.5 BIPs: valori e intervalli

Come già detto precedentemente, il processo di classificazione ha coinvolto diversi dati tra cui i breaks in presence. Questo attributo è stato utilizzato in una parte delle classificazioni come semplice attributo e in altre come classe. Nel primo caso esso può concorrere a definire il modello che descrive l'attributo classe; nel secondo caso è esso stesso la classe, quindi si cerca di definirlo in termini degli altri attributi. In entrambi i casi questi dati sono stati considerati sia come variabili numeriche, sia come variabili nominali. Come già spiegato in dettaglio nel capitolo 3, le variabili nominali possono assumere un numero finito di valori. La scelta di ricorrere a questo tipo di dato è nata dalla particolarità dei dati forniti dagli esperti di realtà virtuale. Ricordiamo infatti che l'esperimento è stato condotto con solo quaranta partecipanti e il numero di BIPs riportati da ciascuno di essi è distribuito su una scala da [0,16].

Può presentarsi in questo caso una situazione nella quale si vogliono definire delle regole di decisione per predire il numero di BIPs. Suddividendo i BIPs in intervalli, è possibile individuare delle regole generali valide per un numero di BIPs compreso in un certo range. La confidenza associata a regole estratte con tale espediente sarà decisamente più alta rispetto al caso in cui la classe BIPs venga considerata come una variabile numerica. Il numero di partecipanti che hanno un numero di BIPs compreso in un intervallo è maggiore o al più uguale al numero di partecipanti che segnalano un numero di BIPs pari ad uno solo dei valori compresi nel range.

Si è scelto quindi di condurre delle classificazioni sia con l'attributo BIPs definita come una variabile numerica, sia definita come una variabile nominale. In quest'ultimo caso alla variabile nominale sono stati associati dei valori che rappresentano un intervallo. Si sono considerati sia intervalli regolari che intervalli irregolari. Gli intervalli regolari hanno tutti la stessa dimensione, quelli irregolari hanno dimensioni diverse. Gli intervalli irregolari sono stati definiti considerando la distribuzione del numero di BIPs.

Nel caso in cui per la rappresentazione dell'attributo BIPs si utilizzi la suddivisione in intervalli regolari, allora ad un soggetto per tale attributo verrà assegnato:

- il valore 0 se il numero di BIPs riportato è compreso nell'intervallo [0,3];
- il valore 1 se il numero di BIPs riportato è compreso nell'intervallo [4,7];
- il valore 2 se il numero di BIPs riportato è compreso nell'intervallo [8,11];
- il valore 3 se il numero di BIPs riportato è compreso nell'intervallo [12,15];
- il valore 4 se il numero di BIPs riportato è compreso nell'intervallo [16, ∞);

Nel caso in cui per la rappresentazione dell'attributo BIPs si utilizzi la suddivisione in intervalli irregolari, allora ad un soggetto per tale attributo verrà assegnato:

- il valore 0 se il numero di BIPs riportato è compreso nell'intervallo [0,1];
- il valore 1 se il numero di BIPs riportato è 2;

- il valore 2 se il numero di BIPs riportato è 3;
- il valore 3 se il numero di BIPs riportato è compreso nell'intervallo [4,5];
- il valore 4 se il numero di BIPs riportato è compreso nell'intervallo [6,16];

6.7.6 Risultati interessanti

L'algoritmo di classificazione utilizzato produce, sulla base del dataset di oggetti fornito in input e della classe da predire, un albero le cui foglie sono etichettate con l'attributo classe, mentre i nodi interni sono etichettati con vincoli sugli attributi non classe. Da un tale albero è possibile estrarre tante regole di decisione quanti sono i possibili percorsi dal nodo radice a qualsiasi nodo foglia. Come già spiegato in dettaglio nel capitolo 3, ogni regola può essere scomposta in due parti, la premessa e la conclusione. Se seguendo un percorso dalla radice dell'albero ad una foglia si individua una regola, allora tutti i vincoli definiti nei nodi interni attraversati dal percorso rappresentano la premessa della regola. L'informazione memorizzata nel nodo foglia rappresenta la conclusione della regola. All'albero prodotto dall'algoritmo di classificazione sono associate due informazioni: *la percentuale di errore associata all'albero* e *la percentuale di errore associata a ciascuna regola derivabile dall'albero*. La percentuale di errore associata all'albero dipende dalla percentuale di errore dei singoli percorsi individuabili in esso e quindi dalle singole regole. La nostra analisi quindi si è concentrata sulle singole regole estraibili.

La classificazione, indipendentemente dall'attributo scelto come target, ha coinvolto tutti o alcuni degli attributi associati ai partecipanti all'esperimento. In alcuni casi la presenza di più attributi può consentire la definizione di modelli più ricchi per la classe. In altri casi, la descrizione degli oggetti del dataset fornito in input realizzata con troppi attributi può portare ad individuare modelli poco interessanti o validi con una probabilità molto bassa. Alcuni attributi

possono compromettere il modello; non è sempre possibile individuare quali siano gli attributi da escludere per ottenere risultati migliori.

Di seguito sono riportate alcune delle regole più interessanti estratte dai vari alberi di decisione. In appendice B sono riportate tutte le regole estratte.

La classificazione realizzata in questa fase ha portato all'individuazione di regole interessanti per la descrizione dei clusters. Per quanto riguarda la predizione del numero di bips, le regole mostrano una confidenza più bassa per cui non è stata evidenziata nessuna regola particolarmente interessante.

La selezione delle regole estratte con classe da predire il cluster di appartenenza è avvenuta in due fasi successive:

- 1) Fra tutte le regole estratte sono state individuate le regole caratterizzate da una confidenza alta. Come valore soglia è stato considerato una confidenza del 50%.
- 2) Dalle regole individuate come nel punto 1) sono state escluse quelle che fornivano una descrizione del cluster in misura inferiore al 30%.

Di seguito sono riportate le regole che abbiamo ritenuto interessanti. Per ciascuna regola è indicato anche il rapporto tra il numero di casi in cui è verificata la premessa e il numero di casi in cui è verificata la premessa, ma non la conclusione.

Nel caso di clusters determinati con il k-means con un $k=5$ le regole ritenute interessanti in termini di confidenza sono le seguenti:

- Se ((Media Gsr ≤ 0.102) & (Media Respirazione ≤ -147070) & (Avatar = realistico) & (Sesso = maschio)) allora cluster di appartenenza = 0 (6.0 / 2.3)
- Se ((Media Hr > 76.053) & (Avatar = realistico) & (Sesso = maschio) & (Media Gsr ≤ 0.038)) allora cluster di appartenenza = 0 (5.0 / 1.2)
- Se ((Media Hr > 76.053) & (Avatar = cartoon) & (Sesso = femmina) & (Media Gsr > -0.017)) allora cluster di appartenenza = 2 (6.0 / 2.3)
- Se ((Avatar = cartoon) & (4 \leq Numero di bips < 8) & (Media Hr > 80.955)) allora cluster di appartenenza = 2 (4.0/1.2)
- Se ((Avatar = cartoon) & (4 \leq Numero di bips < 5) & (Media Hr > 80.955)) allora cluster di appartenenza = 2 (4.0/1.2)

- Se ((Media Hr <= 76.053) & (Media Respirazione > -142070)) allora cluster di appartenenza = 3 (5.0/1.2)
- Se ((Numero di bips = 5) & (Media Respirazione > -154360)) allora cluster di appartenenza = 3 (3.0/1.1)
- Se ((Avatar = cartoon) & (4 <= Numero di bips < 8) & (Media Hr<=80.955)) allora cluster di appartenenza = 3 (3.0/1.1)
- Se ((Avatar = cartoon) & (4<= Numero di bips<= 5) & (Media Hr<=80.955)) allora cluster di appartenenza = 3 (3.0/1.1)
- Se ((Media Respirazione > -147070) & (-0.008< Media Gsr <= 0.064) & (Texture = bassa)) allora cluster di appartenenza= 3 (4.0/1.2)
- Se (Media Gsr > 0.102) allora cluster di appartenenza = 4 (6.0/2.3)

Nel caso di clusters determinati con il k-means con un k= 6 l'unica regola che abbiamo ritenuto interessante è la seguente:

- Se ((Texture = alta) & (Media Gsr > - 0.0434) & (-250062 < Media Respirazione <= -11852) & (76.053< Media Hr <= 109.896)) allora cluster di appartenenza = 1 (7.0/2.3)

Nel caso di clusters determinati con l'algoritmo gerarchico con k= 5 le regole ritenute interessanti sono le seguenti:

- Se ((Sesso = femmina) & (Media Hr <= 87.689) & (0.002 < Media Gsr <= 0.063)) allora cluster di appartenenza = 2 (4.0/1.2)
- Se ((Sesso= 'maschio') & (Avatar= realistico) & (Texture= bassa)) allora cluster di appartenenza = 3 (5.0/1.2)
- Se ((0<=Numero di bips < 4) & (Media Respirazione<= -136168) & (Sesso= maschio)) allora cluster di appartenenza= 3 (8.0/3.5)

Nel caso di clusters determinati con l'algoritmo gerarchico con un k= 6 l'unica regola ritenuta interessante è la seguente:

- Se ((Sesso= 'maschio') & (Avatar= realistico) & (Texture= bassa)) allora cluster di appartenenza = 4 (5.0/1.2)

L'interpretazione di queste regole è facile da realizzare. Considerando ad esempio l'ultima regola riportata, risulta che su 5 partecipanti di sesso maschio

che hanno navigato in un ambiente virtuale caratterizzato da una qualità grafica bassa e popolato da avatars realistici, 3.8 di essi appartengono al cluster 4. Il numero di elementi appartenenti al cluster 4 non è un valore intero. Questo è dovuto alla fase di pruning che fa una stima dell'errore intrinseco ai dati. Nell'ipotesi di dati corretti, la fase di pruning non sarebbe stata necessaria e il numero di soggetti che apparterrebbero al cluster 4, soddisfacendo le premesse, sarebbe stato maggiore di 3.8, presumibilmente 4.

6.8 Ottimizzazioni

Un'analisi dei risultati ottenuti con l'approccio seguito nella prima fase ha evidenziato l'inadeguatezza di alcune scelte che possono aver compromesso i risultati del processo di classificazione. Alcune riguardano l'utilizzo in fase di classificazione di attributi non necessari, altre riguardano i valori considerati per i parametri fisiologici. Nei sottoparagrafi che seguono è fornita una descrizione delle anomalie riscontrate negli attributi utilizzati in fase di classificazione e le soluzioni adottate per ottenere dei risultati migliori.

6.8.1 Eliminazione attributi

Nella prima fase dell'analisi, la classificazione è stata realizzata considerando parte dell'intero set di attributi. Alcune classificazioni sono state fatte considerando per ciascun soggetto solo alcuni attributi. Si è resa necessaria una tale strategia perché non è detto che considerare più attributi porti a ricavare regole di decisione migliori. Perché si possano individuare regole interessanti è necessario classificare per passi successivi: inizialmente si classifica considerando tuple definite con pochi attributi e poi nelle classificazioni successive si vanno ad aggiungere altri attributi fino a comprenderli tutti. Questo è l'approccio che abbiamo seguito. In alcuni casi la

presenza di un attributo può falsare il risultato; una tale situazione si verifica se l'attributo è poco significativo. E' questo il caso di uno degli attributi considerati nella prima fase della nostra analisi: la media della respirazione. Diversamente dagli altri parametri fisiologici, l'informazione relativa alla respirazione in questo contesto è poco interessante e può compromettere la validità delle regole di decisione. Il suo utilizzo nella fase di classificazione è scaturito dall'esigenza di trovare una correlazione con le traiettorie compiute dai soggetti. Una delle ipotesi da verificare è se i soggetti che hanno una traiettoria simile, hanno anche dei parametri fisiologici simili. Il problema che si presenta in questo contesto è dovuto al fatto che non è significativo confrontare due soggetti rispetto alla respirazione. La respirazione assume dei valori che non dipendono da situazioni che influenzano lo stato d'animo del soggetto.

La sua variabilità, infatti, dipende dalle condizioni di salute del soggetto, ad esempio se ha o meno problemi respiratori o se compie attività sportiva.

La sudorazione e l'heart rate invece dipendono fortemente dallo stato d'animo del soggetto. Una situazione come quella che si viene a creare in un ambiente virtuale può essere molto forte e può creare nel soggetto uno stato di ansia e di stress. Un tale stato psicologico si ripercuote nella sudorazione e nell'heart rate ma non nella respirazione. Poiché ciò che interessa è verificare se un determinato ambiente produca uno stesso effetto su vari soggetti, si è deciso di trascurare l'attributo relativo alla respirazione e di considerare come parametri fisiologici solo sudorazione e heart rate.

6.8.2 Normalizzazione parametri fisiologici

In questa seconda fase di analisi si è scelto di considerare tra i parametri fisiologici solo la sudorazione e l'heart rate. Nella fase precedente di questi valori è stato considerato il valore medio riscontrato durante la navigazione. Questo ha richiesto l'implementazione di un algoritmo per estrarre i dati necessari al calcolo di questa informazione. Un'attenta riflessione ci ha portato

però a ritenere poco significativo questo valore. Infatti può non essere completamente corretto considerare i valori per la sudorazione calcolato in questo modo. Seguire un percorso virtuale non è equivalente a percorrere una strada di una città reale. Durante un esperimento di realtà virtuale entrano in gioco diversi fattori quali ansia e stress che influenzano i parametri fisiologici.

Se in una situazione reale ha senso calcolare il valore medio di sudorazione o heart rate in un certo intervallo di tempo, quando il soggetto si trova in una situazione anomala, bisogna utilizzare alcuni accorgimenti. Per valutare questi parametri fisiologici durante l'esperimento di realtà virtuale bisogna tener conto anche dei valori di questi parametri in una situazione normale.

Per la determinazione dei valori da associare agli attributi relativi ai due parametri fisiologici sono stati utilizzati due approcci diversi. In entrambi i casi si sono però considerati i valori rilevati durante l'esperimento. Questi valori sono stati messi a confronto con i valori rilevati durante la *baseline*, una delle fasi preliminari alla navigazione; tali valori sono espressione di quelli che un soggetto ha normalmente.

In fase di classificazione l'attributo associato a ciascun soggetto relativo all'heart rate rappresenta la variazione registrata durante la navigazione. La variazione dell'heart rate è stata calcolata come la differenza tra la media delle pulsazioni rilevata in una situazione reale e quella rilevata durante la navigazione. Per quanto riguarda la sudorazione si è considerata la variazione percentuale. Questo ha richiesto per ciascun soggetto:

- il calcolo del valore medio della sudorazione in una situazione normale (durante la baseline) m_b ;
- la sostituzione di ogni valore v della sudorazione rilevato in fase di navigazione con il valore

$$v' = (v - m_b) / fabs(m_b), \quad (6.1)$$

dove $fabs(m_b)$ rappresenta il valore assoluto della media ;

- il calcolo del valor medio dei v' .

Il calcolo della variazione percentuale di sudorazione richiede l'implementazione di un algoritmo che esegue i seguenti passi:

- 1) Per ciascun soggetto i si memorizza l'istante di tempo in cui inizia la fase di baseline e quello in cui termina. Indichiamo queste informazioni rispettivamente con $start_baseline(i)$ ed $end_baseline(i)$. Si memorizza inoltre l'istante di tempo in cui inizia la fase di navigazione e quella in cui termina. Indichiamo queste due informazioni con $start(i)$ ed $end(i)$. Questi dati sono memorizzati nel file *timing* che per ciascun soggetto riporta gli istanti di tempo in cui hanno inizio le varie fasi dell'esperimento.

- 2) Per ciascun soggetto i si calcola la media della sudorazione m_b durante la baseline.

Un valore v per questo parametro misurato all'istante t contribuisce al calcolo della media m_b se $start_baseline(i) \leq t < end_baseline(i)$.

- 3) Per ciascun soggetto si calcola la media dei valori v di sudorazione rilevati durante la navigazione. Nel calcolo della media si considerano i valori v normalizzati, indichiamoli con v' . Il valore normalizzato di v è calcolato utilizzando (6.1).

Una misurazione v contribuisce al calcolo della media e di conseguenza viene normalizzato se l'istante di tempo t in cui è stato rilevato è tale che $start(i) \leq t < end(i)$.

La media così calcolata per ciascun soggetto, rappresenta l'attributo da utilizzare per indicare il valore del parametro fisiologico durante la navigazione.

6.8.3 Risultati interessanti

L'approccio descritto ha permesso l'individuazione di regole abbastanza interessanti e interpretabili da un punto di vista semantico. La validità di queste regole è legata sia alla confidenza associata, sia alla coerenza con i risultati ottenuti dagli esperti di realtà virtuale con strumenti statistici tradizionali.

Nell'approccio seguito ci si è limitati a considerare i clusters determinati con l'algoritmo gerarchico. Questa scelta è scaturita dalla similarità tra i clusters determinati con il k-means e quelli determinati con il gerarchico. E' stato preferito quest'ultimo per le buone prestazioni offerte in caso di applicazione su pochi campioni, come nel nostro caso. Inoltre si è assunto che i dati di input fossero corretti, per cui non è stata realizzata l'operazione di pruning su di essi.

La selezione delle regole ritenute più interessanti è stata realizzata in questo modo:

- Se l'attributo target è il cluster di appartenenza, si considerano le regole caratterizzate da un buon livello di confidenza tenendo conto anche della cardinalità del cluster. Se il processo di classificazione consente di estrarre una regola valida con una confidenza del 40%, tale regola è ritenuta interessante se il numero di elementi appartenenti al cluster è alto.
- Se l'attributo target è il numero di bips e tra gli attributi che compaiono nella premessa della regola è presente il cluster di appartenenza, l'interesse della regola è valutato considerando, oltre alla confidenza, il numero di elementi del cluster stesso.
- Se l'attributo target è il numero di bips e tra gli attributi che definiscono la premessa della regola non compare il cluster di appartenenza, per valutare l'interesse di una regola si tiene conto del rapporto tra confidenza e numero totale di campioni considerati. Nel caso specifico il numero di campioni è quaranta.

La fase di analisi delle regole estratte con il processo di classificazione, ci ha portato ad individuare quelle più interessanti.

Nel caso di clusters determinati con $k=5$ e con attributo target il cluster, sono state individuate le seguenti regole:

- Se $((0 \leq \text{numerodiBIPs} \leq 3) \ \& \ (\text{sex} = \text{maschio}) \ \& \ (\text{Variation} \leq -0.18) \ \& \ (\text{Avatar} = \text{realistico}))$ allora cluster= 3 (4.0)
- Se $((\text{GameTimes} = 1) \ \& \ (\text{Programming} = 1) \ \& \ (\text{Games} = 1))$ allora cluster= 4 (8.0/2.0)

Quest'ultima regola risulta particolarmente interessante. Il basso valore degli attributi *GameTimes*, *Programming* e *Games* delinea un gruppo di volontari poco propenso al gioco e all'utilizzo del computer in generale, quindi con poche conoscenze o esperienze di grafica tridimensionale. Il raggruppamento in un unico cluster, con un basso margine di errore, porta a pensare che abbiano un approccio simile durante l'esplorazione dell'ambiente virtuale.

Nel caso di regole di regole determinate con attributo target il cluster di appartenenza e con i clusters determinati con l'algoritmo gerarchico per $k=6$, risulta particolarmente interessante la seguente regola:

- Se $((0 \leq \text{numerodiBIPs} \leq 3) \& (\text{Avatar} = \text{realistico}) \& (\text{Sesso} = \text{maschio}) \& (\text{VariazHr} \leq -0.209))$ allora cluster= 4 (4.0)

Questa regola pone in evidenza come ci sia un forte legame tra il livello di presenza e le condizioni *(Avatar= realistico)* e *(Sesso= maschio)*. I soggetti di sesso maschio immersi in un ambiente popolato da avatar realistici risultano altamente presenti nell'ambiente. Questa ipotesi è ulteriormente confermata dalla variazione dell'heart rate che risulta molto bassa. L'individuazione di questa regola va a confermare e completare le analisi già condotte dagli esperti di realtà virtuale e può fornire delle indicazioni per eventuali analisi successive.

Le regole ritenute più interessanti nel caso in cui il target sia rappresentato dal numero di bips sono riportate di seguito.

Nel caso della suddivisione in k clusters con $k=5$:

- Se $(\text{cluster} = 1)$ allora $0 \leq \text{numerodiBIPS} \leq 1$ (3.0/1.0)
- Se $(\text{cluster} = 1)$ allora $0 \leq \text{numerodiBIPS} \leq 3$ (3.0)
- Se $(\text{Age} = 18)$ allora $0 \leq \text{numerodiBIPS} \leq 3$ (3.0)

Nel caso della suddivisione in k clusters con $k=6$:

- Se $((\text{cluster} = 1) \& (\text{VariazGSR} \leq 3.14))$ allora $\text{numerodiBIPs} = 5$ (2.0)
- Se $((\text{cluster} = 1) \& (\text{VariazGSR} \leq 3.14))$ allora $4 \leq \text{numerodiBIPs} \leq 5$ (2.0)
- Se $((\text{cluster} = 1) \& (\text{VariazGSR} \leq 3.14))$ allora $4 \leq \text{numerodiBIPs} \leq 7$ (2.0)

-
- Se (cluster= 2) allora $0 \leq \text{numerodiBIPs} \leq 1$ (3.0/1.0)
 - Se (cluster= 2) allora $0 \leq \text{numerodiBIPs} \leq 3$ (3.0)
 - Se (Age= 18) allora $0 \leq \text{numerodiBIPs} \leq 3$ (3.0)
 - Se ((Age= 19) & (VariazHr ≤ -0.3)) allora $0 \leq \text{numerodiBIPs} \leq 3$ (2.0)
 - Se ((Age= 19) & (VariazHr > -0.3)) allora $4 \leq \text{numerodiBIPs} \leq 5$ (2.0)
 - Se (Age= 20) allora $0 \leq \text{numerodiBIPs} \leq 3$ (3.0/1.0)
 - Se (Age= 21) allora $0 \leq \text{numerodiBIPs} \leq 3$ (3.0/1.0)
 - Se ((Age= 24) & (VariazGSR ≤ 0.616)) allora $0 \leq \text{numerodiBIPs} \leq 1$ (2.0)
 - Se (Age= 25) allora $0 \leq \text{numerodiBIPs} \leq 3$ (3.0/1.0)
 - Se (Age= 27) allora $\text{numerodiBIPs} = 5$ (2.0)
 - Se (Age= 28) allora $4 \leq \text{numerodiBIPs} \leq 7$ (3.0/1.0)
 - Se (Age= 30) allora $\text{numerodiBIPs} = 3$ (2.0)

Tutte queste regole che consentono di predire il numero di BIPs in funzione degli altri attributi risultano molto interessanti perché la loro interpretazione si basa sulla stretta relazione esistente tra numero di BIPs e livello di presenza. Il numero di BIPs riportati dai diversi soggetti durante l'esperimento varia da 0 a 16, con una media di 3.6 e una varianza di 8.3. Si può definire basso un numero di BIPs compreso tra 0 e 3.

Dalle regole estratte, il cluster 1 nel caso $k=5$ e il cluster 2 nel caso $k=6$ risultano gruppi con un livello alto di presenza. Il 100% degli elementi di questi gruppi ha riportato un numero di BIPs inferiore a 4, quindi basso. Questi risultati sono interessanti in quanto ci consentono di identificare uno specifico gruppo di persone che ha avuto un alto livello di presenza e può rappresentare il punto di partenza per un'analisi successiva di individuazione di altre correlazioni. Nel caso specifico l'informazione sul cluster di traiettorie può essere utilizzata per confrontarla con il cluster delle persone che sono risultate meno presenti e per verificare se ci sono differenze significative nel tipo di traiettoria compiuta.

Una considerazione particolare è richiesta per le regole che nella premessa contengono un vincolo 'Age= 19'. In entrambe le regole compare un vincolo

sull'heart rate. Questo pone in evidenza e conferma le ipotesi degli esperti di realtà virtuale sul forte legame esistente tra heart rate e livello di presenza. Con una bassa variazione di Heart rate, quindi maggiore rilassamento, si ha un numero basso di BIPs, quindi maggiore presenza. Al contrario, con una maggiore variazione di heart rate, quindi maggiore stress, si ha un numero maggiore di BIPs, quindi minore presenza.

Questa deduzione viene fatta sulla base di 4 campioni, quindi non è affidabile dal punto di vista statistico. Tuttavia il legame tra Heart rate e Presence è ben documentato, quindi verificabile. In questo caso, possiamo dire che la regola rispetta ipotesi note e comprovate, quindi è potenzialmente utilizzabile in un ambito più ampio considerando un dataset più ricco di dati.

Un altro elemento interessante da valutare, in analisi future, è il forte legame che sembra esserci tra età e numero di BIPs, quindi tra età e il livello di presenza. Le regole ottenute suggeriscono un legame tra la bassa età dei volontari (< 21) e l'alto valore di Presence, ma il basso numero di elementi nei gruppi non autorizza alcuna conclusione definitiva. In appendice C sono riportate tutte le regole estratte.

6.9 Clustering: Approccio Alternativo

La soluzione proposta con questo lavoro di tesi per l'applicazione del data mining alla realtà virtuale prevede una prima fase di clustering che consente di individuare gruppi di soggetti che compiono durante la navigazione una traiettoria simile. La seconda fase dell'approccio consiste nell'utilizzo di strumenti di classificazione sugli attributi associati ai soggetti, per individuare delle regole che descrivono quelli che compiono una stessa traiettoria, o quelli che hanno riportato lo stesso numero di BIPs. L'approccio seguito è dipeso dal tipo dei dati estratti dall'esperimento oggetto del nostro studio. La disponibilità

di altri dati avrebbe permesso l'estrazione di informazioni diverse e reso necessario l'utilizzo di un altro approccio per realizzare l'analisi.

Di seguito è proposto un approccio alternativo all'analisi dei dati estratti durante l'esperimento di realtà virtuale. Come già detto nel capitolo 5, in cui viene descritto l'esperimento, l'ambiente ricreato dagli esperti della University College London è la riproduzione di un centro commerciale. In esso sono presenti una serie di negozi di vario tipo: di articoli sportivi, calzature ed altri. Durante la navigazione ciascun partecipante poteva visitare ciascun negozio o osservarne semplicemente la vetrina. Sarebbe interessante studiare le reazioni dei vari soggetti in prossimità di un negozio di un certo tipo.

Se un approccio di questo tipo fosse applicabile si fornirebbe un grande contributo al data mining come strumento in ambito economico. Le tecniche di data mining vengono infatti utilizzate per le analisi del mercato e per lo studio delle abitudini di acquisto dei clienti. La realtà virtuale in questo caso può risultare di grande ausilio. Può essere interessante, ad esempio, verificare se soggetti appartenenti ad una stessa fascia di età in prossimità di un certo negozio hanno la stessa reazione. Se da un'analisi dei dati si constata che in prossimità di un negozio di calzature i soggetti di età compresa tra i 20 e 25 risultano attratti dalla vetrina (presentano una variazione significativa nei valori dei parametri fisiologici), può essere di stimolo per i rivenditori di calzature perché si specializzino nella vendita di articoli per giovani. In alternativa possono essere condotte campagne promozionali per questi potenziali clienti. L'esperimento oggetto della nostra analisi si presta ad uno studio di questo tipo. Si conoscono le posizioni e i parametri fisiologici dei partecipanti ad intervalli regolari. Si conosce inoltre la dislocazione dei negozi nell'ambiente ricreato. L'obiettivo in questo caso è l'individuazione di gruppi di partecipanti che oltre ad avere una traiettoria simile hanno anche un andamento nei parametri fisiologici simile. Lo scopo è quello di trovare gruppi di partecipanti

che in prossimità di un negozio di un certo tipo hanno lo stesso comportamento in termini di traiettoria e reazione emotive.

Di uno studio di questo tipo viene fornita solo una linea guida. Sui dati estratti fino ad ora dall'esperimento per ciascun partecipante si conosce la posizione ad ogni istante, ma non si conosce l'orientamento (se a nord, sud, ovest o est). L'individuazione di una tale informazione richiede parecchio tempo, non essendo fornita in maniera esplicita dai dispositivi utilizzati per il rilevamento della posizione.

L'approccio proposto prevede due fasi:

- 1) Individuazione di gruppi di soggetti che risultano simili per traiettoria e parametri fisiologici;
- 2) Definizione di regole per la descrizione di ciascun gruppo definito come nel punto 1).

La prima fase viene realizzata ricorrendo ad un algoritmo di clustering, mentre la seconda fase richiede l'utilizzo di tecniche di classificazione.

6.9.1 Algoritmo di clustering: Calcolo similarità

L'algoritmo di clustering, utilizzato per individuare gruppi di partecipanti simili in termini di traiettorie e parametri fisiologici, è una versione specifica per il trattamento di traiettorie. La similarità tra gli oggetti è valutata in termini di distanza euclidea.

Come già spiegato in dettaglio nel capitolo 3, la distanza euclidea tra due oggetti p -dimensionali $\mathbf{i} = (x_{i1}, x_{i2}, \dots, x_{ip})$ e $\mathbf{j} = (x_{j1}, x_{j2}, \dots, x_{jp})$ è calcolata con la formula 3.3.

L'algoritmo utilizzato considera ogni traiettoria definita come una sequenza di n punti. Ogni punto della traiettoria è definito in uno spazio bidimensionale. La variante utilizzata nell'algoritmo per il calcolo della distanza tra due traiettorie prevede l'utilizzo della formula 3.3 con $p=2$, ovvero

$$d(i, j) = \sqrt{\left(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 \right)} \quad (6.2)$$

dove x_{i1} e x_{i2} sono rispettivamente l'ascissa e l'ordinata della posizione del soggetto i ad un certo istante. Ad ogni iterazione il calcolo della distanza tra traiettorie viene fatta considerando la distanza tra punti rilevati allo stesso istante di tempo.

Il nostro approccio all'analisi dei dati di realtà virtuale richiede che tale algoritmo di clustering, in fase di calcolo della distanza tra due traiettorie tenga conto non solo della distanza tra le posizioni assunte dai soggetti in un certo istante di tempo, ma anche della distanza tra i soggetti in termini di parametri fisiologici. Un'analisi di questo tipo porta ad individuare clusters tali che soggetti appartenenti allo stesso gruppo siano simili sia in termini di traiettoria che di parametri fisiologici. Un tale esigenza ha richiesto la modifica della formula della distanza euclidea utilizzata nell'algoritmo per tener conto anche delle misurazioni di sudorazione e di respirazione. In questo caso un soggetto è rappresentato con una variabile p -dimensionale con $p=4$.

La formula per il calcolo della distanza euclidea diventa

$$d(i, j) = \sqrt{\left(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + |x_{i3} - x_{j3}|^2 + |x_{i4} - x_{j4}|^2 \right)} \quad (6.3)$$

se il soggetto i è rappresentato dalla quadrupla $i = (x_{i1}, x_{i2}, x_{i3}, x_{i4})$. La prima e la seconda variabile indicano rispettivamente ascissa e ordinata della posizione ad un certo istante, la terza e la quarta indicano rispettivamente i valori di sudorazione e respirazione allo stesso istante.

6.9.2 Clustering: Normalizzazione dei dati

Il calcolo della similarità tra due entità p-dimensionali (nel nostro caso $p=4$) con l'utilizzo della formula 6.3 è corretta se le variabili che descrivono i soggetti sono dello stesso tipo. Se l'ascissa e l'ordinata di un punto sono espresse in uno stesso sistema di riferimento non si può dire altrettanto per i valori di sudorazione e respirazione. Le coordinate spaziali e i parametri fisiologici sono espressi in unità di misura diverse; perché il calcolo della distanza euclidea con l'utilizzo della formula 6.3 sia corretta è necessario rendere omogenee le quattro variabili. E' necessario definire i quattro attributi nello stesso sistema di riferimento affinché siano confrontabili.

Per realizzare una tale normalizzazione sono stati utilizzati due approcci diversi:

- *Min-Max normalization;*
- *Z-score normalization.*

Il *Min-max Normalization* prevede che i valori di tutti e quattro gli attributi che descrivono gli oggetti siano mappati su uno stesso intervallo. Indichiamo gli estremi di questo intervallo new_min_A e new_max_A . Per ciascun attributo A i cui valori devono essere normalizzati, si calcola il valore minimo e il valore massimo rilevato per esso. La ricerca riguarda il valore dell'attributo per tutti i soggetti. Indichiamo questi due valori con min_A e max_A rispettivamente. Terminata tale ricerca, ogni valore v dell'attributo A rilevato per ciascun soggetto viene sostituito con il nuovo valore v' calcolato in questo modo:

$$v' = \frac{v - \min_A}{\max_A - \min_A} (new_max_A - new_min_A) + new_min_A \quad (6.4)$$

L'intervallo $[new_min, new_max]$ è per tutti gli attributi l'intervallo $[0,1]$.

Lo scopo infatti è quello di mappare i valori per i quattro attributi nello stesso intervallo per renderli confrontabili.

Lo z-score normalization prevede che per ciascun attributo venga considerata la media dei valori che assume (definiti su tutti i soggetti) e la deviazione standard. Indichiamo la media e la deviazione standard dell'attributo A con \bar{A} e σ_A rispettivamente.

Ricordiamo che la media degli n valori per un attributo A si calcola come

$$\bar{A} = \frac{\sum_{i=1}^n v_i}{n} \quad (6.5)$$

La deviazione standard invece si calcola come:

$$\sigma_A = \sqrt{\frac{\sum_{i=1}^n (v_i - \bar{A})^2}{n}} \quad (6.6)$$

Determinato il valor medio e la deviazione standard di ciascun attributo considerato (quindi coordinata lungo l'asse x e lungo l'asse z, valore di sudorazione e respirazione), si sostituiscono i valori v di ciascun attributo con

$$v' = \frac{(v - \bar{A})}{\sigma_A} \quad (6.7)$$

In generale la normalizzazione realizzata con lo z-score normalization offre migliori risultati nel caso tra i valori dei dati siano presenti outliers. La nostra scelta è stata quella di utilizzare comunque entrambi i metodi di normalizzazione. Questo ha richiesto l'applicazione del clustering ai dati rappresentati in due modi diversi. I clusters definiti nei due casi risultano molto diversi.

Una descrizione dettagliata dei due metodi di normalizzazione è fornita in [Han00].

6.9.3 Clustering: organizzazione del dataset di input

L'approccio proposto prevede che il clustering venga realizzato considerando sia le traiettorie compiute durante la navigazione sia i valori per i parametri

fisiologici. Il file da fornire in input all'algoritmo di clustering dovrà avere una forma simile a quella riportata in figura 6.9.

Ide₁	t₁¹	X₁¹	Z₁¹	R₁¹	S₁¹
Ide₁	t₁²	X₁²	Z₁²	R₁²	S₁²
.....
Ide₁	t₁ⁿ	X₁ⁿ	Z₁ⁿ	R₁ⁿ	S₁ⁿ
Ide₂	t₂¹	X₂¹	Z₂¹	R₂¹	S₂¹
.....
Ide_m	t_m¹	X_m¹	Z_m¹	R_m¹	S_m¹
Ide_m
Ide_m	t_m^k	X_m^k	Z_m^k	R_m^k	S_m^k

Figura 6.9 Organizzazione del file di input per l'algoritmo di clustering nell'approccio alternativo

Nel file di input i dati devono essere organizzati su sei colonne. Ogni riga contiene quindi sei informazioni: l'identificativo della traiettoria, l'istante di tempo in cui sono state rilevate posizione e parametri fisiologici, la coordinata spaziale lungo l'asse x e lungo l'asse z, il valore rilevato per la respirazione e quello della sudorazione. Tra i records del file è definito un ordinamento. Il primo attributo rispetto al quale si ordinano i dati è l'identificatore. Per ciascuna traiettoria un secondo ordinamento viene fatto in base al tempo.

6.9.4 Creazione del file di input per il clustering: algoritmo

La creazione del file da fornire in input all'algoritmo di clustering richiede l'implementazione di un algoritmo. E' richiesto un tale approccio perché i dati relativi a respirazione, sudorazione, coordinate spaziali lungo l'asse x e lungo l'asse z, sono state organizzate dagli esperti di realtà virtuale in files differenti. Una descrizione dell'organizzazione di questi file è fornita in appendice A.

La procedura per la creazione di un unico file contenente queste informazioni, consistente con quello richiesto dall'algoritmo di clustering, si articola nei seguenti passi:

1. Normalizzazione dei valori di respirazione, sudorazione, coordinate lungo l'asse x e coordinate lungo l'asse z. Questa operazione richiede l'accesso a quattro files diversi. La normalizzazione è realizzata con lo z-score normalization o con il min-max normalization.
2. Si accede al file timing che, per ciascun partecipante, tiene traccia degli istanti di tempo in cui iniziano le varie fasi dell'esperimento. Per ciascun partecipante si individuano gli istanti di tempo in cui inizia e termina la fase di navigazione. Indichiamo gli istanti di tempo in cui il soggetto i inizia e termina la navigazione con $start(i)$ ed $end(i)$.
3. Si tiene traccia del tempo tramite una variabile che ad ogni rilevazione viene incrementata di un trentaduesimo. La posizione nello spazio e i parametri fisiologici vengono infatti rilevati ogni trentaduesimo di secondo. Indichiamo questa informazione con $time$.
4. Si accede contemporaneamente ai quattro files contenenti i valori (normalizzati) di interesse e si crea un nuovo file. In esso si memorizzano identificatore, time, coordinata lungo l'asse x e coordinata lungo l'asse z, valore di respirazione e valore di sudorazione. Tale operazione viene fatta per il partecipante i se e solo se $start(i) \leq time < end(i)$. L'accesso contemporaneo ai quattro files è possibile in quanto hanno la stessa struttura e organizzazione dei dati.
5. Ordinamento del file creato per identificatore e time.

6.9.5 Risultati interessanti

I clusters sono stati determinati considerando sia i valori normalizzati con il metodo *min-max*, sia quelli normalizzati con lo z-score. Questo ha portato ad ottenere due suddivisioni in clusters diverse. L'analisi dell'output dell'algoritmo

di clustering ci ha portato a considerare interessanti in entrambi i casi i clusters ottenuti con $k=5$ e $k=6$. In appendice D è riportata una tabella in cui per ciascun partecipante è indicato il cluster di appartenenza nei due diversi casi. La fase successiva di classificazione ci ha consentito di estrarre alcune regole interessanti.

Nel caso di valori normalizzati con *min-max normalization*, per $k=5$ e con attributo target il cluster di appartenenza :

- Se $((Media_HR > 84.81) \ \& \ (sesso= \text{maschio}) \ \& \ (avatar= \text{realistico}))$ allora $numero_cluster=1$ (6.0/2.3)
- Se $(Age=24)$ allora $numero_cluster=1$ (4.0/1.2)
- Se $((76.05 < Media_HR \leq 84.81) \ \& \ (texture= \text{alta}))$ allora $numero_cluster=3$ (3.0/1.1)

Nel caso $k=6$ l'unica regola che abbiamo ritenuto fosse interessante è :

- Se $(Age=24)$ allora $numero_cluster=1$ (4.0/1.2)

Nel caso $k=5$, con attributo target il numero di bips:

- Se $((84.81 \leq Media_HR < 93.13) \ \& \ (texture= \text{alta}))$ allora $(4 \leq numero_BIPs \leq 7)$ (6.0/1.2)
- Se $(Media_HR > 93.13)$ allora $(0 \leq numero_BIPs \leq 3)$ (12.0/5.7)

Nel caso in cui i dati siano normalizzati utilizzando lo z-score normalization, si ottiene una regola interessante solo nel caso in cui l'attributo classe è il cluster di appartenenza e con una suddivisione in k clusters con $k=6$.

La regola ritenuta interessante è :

- Se $(89.62 < Media_HR \leq 108.38)$ allora $numero_cluster=5$ (15.0/6.8)

In appendice D sono riportate tutte le regole estratte.

7 Conclusioni

Questa tesi si è occupata di analizzare con tecniche di data mining dati estratti da un esperimento di realtà virtuale.

Lo studio delle traiettorie compiute dai diversi soggetti durante la navigazione, realizzato con strumenti di clustering, ha permesso l'individuazione di gruppi di soggetti caratterizzati da una traiettoria simile. La similarità tra le traiettorie è stata valutata in termini di distanza euclidea definita sui punti di ciascuna traiettoria. Lo studio delle traiettorie è stato raffinato ulteriormente, mediante l'applicazione di tecniche di classificazione su questi dati. Tramite queste ultime è stato possibile fornire una descrizione di ciascun cluster in termini di altri attributi associati a ciascun soggetto. Questo ha richiesto la ricerca di similarità tra i soggetti appartenenti allo stesso cluster definita in termini di altri attributi come sesso, età, occupazione, parametri fisiologici. Questa analisi ha consentito l'individuazione di regole di decisione che descrivono ciascun cluster. Queste ultime possono essere utilizzate per predire il cluster di appartenenza di un soggetto di cui si conoscono tutte le informazioni eccetto il cluster di appartenenza.

Nell'ipotesi di avere una regola del tipo “Se $età = 23$ & sesso = *femmina* & $Variaz_GSR > 0.1$ allora cluster = 4”, se un soggetto soddisfa le premesse della regola allora apparterrà al cluster 4.

L'utilizzo di queste tecniche ha consentito anche la definizione di regole per la predizione del numero di BIPs; in questo caso si cercano delle affinità in termini dei valori degli altri attributi tra i soggetti che hanno lo stesso numero di BIPs.

Le regole non sono valide in assoluto ed è proprio per questo motivo che la fase finale del nostro studio ha richiesto una selezione delle regole più

interessanti tra quelle estratte. La scelta è stata fatta sulla base della confidenza di ciascuna regola. Nonostante il numero esiguo di traiettorie considerate, le regole ritenute più interessanti hanno associata una buona confidenza. La considerazione di più traiettorie comunque avrebbe permesso una fase di clustering più accurata e l'estrazione, nella fase successiva, di regole più valide.

Maggiore è la quantità di dati che si considerano e migliore è l'analisi prodotta con l'ausilio di strumenti di data mining. Con questo lavoro è stata sviluppata una metodologia innovativa per l'applicazione di data mining a dati di realtà virtuale. Lo studio condotto in questa tesi rappresenta solo l'inizio di una lunga serie di analisi. Per capire quanto questa metodologia possa essere ulteriormente sviluppata è necessario considerare grandi volumi di dati estratti da altri esperimenti e applicare lo stesso procedimento seguito in questa tesi. Infatti, prima di cimentarsi nello sviluppo di una nuova tecnica di data mining per il trattamento dei dati "virtuali" è necessario fare un primo passo di analisi con tecniche di data mining più o meno tradizionali e capire se siano applicabili a quelli estratti da un esperimento specifico.

Appendice A

L'esperimento di realtà virtuale, oggetto del nostro studio, ha prodotto una grossa quantità di dati.

Nella nostra analisi hanno rivestito un ruolo centrale le traiettorie compiute da coloro che hanno partecipato all'esperimento. La ricostruzione delle traiettorie è stata realizzata considerando la stanza, sulle cui pareti è stato proiettato l'ambiente virtuale, come un piano bidimensionale. Su tale piano l'asse x rappresenta l'asse delle ascisse e l'asse z quella delle ordinate.



Figura A.1 Mapping del centro commerciale virtuale su un piano bidimensionale

Come già spiegato in maniera dettagliata nel paragrafo 6.3, la ricostruzione delle traiettorie compiute nell'ambiente virtuale ha richiesto l'elaborazione di due files diversi, *pos_x* e *pos_z*. Questi due files contengono rispettivamente le coordinate lungo l'asse x e lungo l'asse z delle posizioni assunte dai diversi partecipanti durante l'esperimento. Le coordinate spaziali sono state rilevate ad intervalli di tempo regolari, ogni trentaduesimo di secondo. La posizione di

ciascun soggetto è stata rilevata sin dalle fasi di preparazione alla navigazione e fino al termine della fase post-navigazione.

All'interno dei files *pos_x* e *pos_z* i dati sono organizzati per colonne: ci sono tante colonne quanti sono i partecipanti all'esperimento.

I dati della generica colonna *i* sono organizzati in questo modo:

- la prima riga contiene un codice che identifica univocamente il soggetto *i*;
- la seconda riga contiene un valore intero che indica il numero di valori significativi tra quelli che lo seguono;
- la riga *j* successiva alla seconda contiene la posizione lungo l'asse x (risp. asse z) del soggetto rilevata all'istante di tempo *j*.

Tra queste informazioni riveste un ruolo cruciale l'informazione contenuta nella seconda riga. Se la rilevazione della posizione del soggetto *i* ha prodotto *m* posizioni, allora il valore memorizzato nella seconda riga della colonna *j*-esima sarà *m*. In questo caso dovranno essere considerate significative, ai fini della determinazione della traiettoria, le coordinate lungo l'asse x (risp. asse z) memorizzate dalla riga 3 alla riga *m* + 3. I valori memorizzati a partire dalla riga *m* + 4 sono valori nulli rappresentati da 0 da non confondere con la coordinata 0 lungo l'asse x (risp. asse z).

La presenza di valori nulli è dovuta all'origine del file di dati: esso è costruito a partire da una matrice che ha tante colonne quanti sono i partecipanti e tante righe quant'è il numero massimo di posizioni rilevate per qualche soggetto.

Le colonne contenenti le coordinate spaziali lungo l'asse x (risp. asse z) dei soggetti che hanno compiuto una traiettoria più corta conterrà delle righe vuote (dalla *m*+3 in poi se *m* è il numero di occorrenze significative). In figura A-2 è riportato uno screenshot relativo ad un frammento del file *pos_x*. La struttura del file *pos_z* è analoga.

12221	21111	21121	21211	21221	22111	22121
23024	19286	16357	14279	18846	16829	16806
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0,066
0	0	0	0	0	0	0,066
0	0	0	0	0	0	0,066
0	0,068	0,176	0,106	0	0	0,066
0	0,068	0,175	0,106	0	0	0,066
0	0,068	0,176	0,106	0	0,001	0,066
0,011	0,068	0,175	0,106	0	0,573	0,066
0,011	0,068	0,175	0,106	0	0,571	0,066
0,011	0,068	0,175	0,106	0,002	0,571	0,066
0,011	0,068	0,175	0,106	0,705	0,571	0,066
0,011	0,068	0,175	0,106	0,703	0,571	0,066
0,011	0,068	0,175	0,106	0,704	0,571	0,066
0,01	0,068	0,175	0,106	0,704	0,571	0,066

Figura A.2 Frammento del file *pos_x*

Tra i files che ci sono stati forniti dagli esperti di realtà virtuale particolarmente interessante è *timing*. Questo file contiene per ciascun partecipante, l'indicazione degli istanti di tempo in cui hanno inizio le varie fasi dell'esperimento. Come già spiegato nel capitolo 5, l'esperimento virtuale si articola in cinque fasi fondamentali:

- fase preliminare;
- *baseline*;
- fase di *training*;
- fase di navigazione;
- fase post-esperimento.

La durata delle diverse fasi può variare da soggetto a soggetto. Ne consegue che le varie fasi iniziano in istanti temporali differenti per i vari soggetti.

Poiché nell'analisi dei dati si è interessati a fasi specifico dell'esperimento, è necessario sapere con precisione quando hanno inizio le diverse fasi. I dati all'interno del file *timing* sono organizzati per colonne. Contiene tante colonne quanti sono i partecipanti coinvolti nell'esperimento, ovvero quaranta.

La colonna *i* è organizzata in questo modo:

- la prima riga contiene l'identificativo del soggetto *i*;
- la seconda riga contiene l'istante di tempo in cui ha inizio la baseline (*start of the baseline*);
- la terza riga contiene l'istante di tempo in cui ha inizio la fase di training (*start of the training session*);
- la quarta riga contiene l'istante di tempo in cui ha inizio la navigazione vera e propria (*start of the actual experiment*);
- la quinta riga contiene l'istante di tempo in cui ha inizio la fase post-esperimento (*dark sky event*);
- la sesta riga contiene l'istante di tempo in cui finisce l'esperimento (*end of the experiment*).

In base a queste informazioni, una fase termina quando inizia la fase successiva.

In figura A-3 è fornito un frammento del file *timing*.

12121	12211	12221	21111	21121	21211	21221	22111	22121	22211	22221
549	2001	546	453	906	477	855	483	471	2211	2394
3432	4887	3429	3336	3789	3360	3738	3366	3354	5094	5274
16353	18627	16857	12993	9126	7872	11973	9978	10200	16251	11235
22116	24390	22623	18753	14886	13632	17736	15741	15960	22014	16995
22524	24570	22800	18936	15414	14112	18201	16185	16452	22653	17448

Figura A-3 Frammento del file *timing*

Tra i files fornitici dagli esperti di realtà virtuale, è risultato particolarmente utile il file *bips*. Questo file contiene il numero di *breaks in presence* riportati da ciascun soggetto e gli istanti di tempo in cui è avvenuta la segnalazione.

All'interno del file i dati sono organizzati per colonne: ci sono tante colonne quanti sono i soggetti presi in considerazione ovvero quaranta.

La colonna *i* è organizzata in questo modo:

- la prima riga contiene un codice che identifica univocamente il soggetto *i*;
- la seconda riga contiene il numero di BIPs riportati dal soggetto *i*.

Indichiamo questo valore con *m*.

- le *m* righe successive contengono l'indicazione sugli istanti in cui occorrono i diversi BIPs.

In figura A-4 è riportato un frammento del file *bips*.

22211	22221	31111	31121	31211	31221	32111	32121
3	4	8	2	5	10	4	2
8637	13938	10665	14239	10037	11169	6003	16770
8856	14694	11370	14674	10916	11676	12942	19023
9048	15918	12423		10928	12039	15417	
	17232	13512		11780	12630	17199	
		14460		14210	13461		
		14676			13680		
		15384			14616		
		16356			15489		
					16230		
					16833		

Figura A-4 Frammento del file *bips*

Il nostro studio ha richiesto l'analisi dei parametri fisiologici dei partecipanti all'esperimento. Queste informazioni sono memorizzate nei files *GSR_det*, *Resp*, *EKG*, che contengono rispettivamente i dati di sudorazione, respirazione e attività cardiaca. L'organizzazione dei dati all'interno di questi files è la stessa di *pos_x* e *pos_z*. I parametri fisiologici, infatti, vengono misurati ogni trentaduesimo di secondo coi come le coordinate geografiche. Dai dati relativi all'attività cardiaca è possibile determinare il numero medio di contrazioni al

minuto che è indicato convenzionalmente nelle regole di decisione estratte come *Media_HR*.

Oltre alle informazioni appena descritte sono stati utilizzati anche i dati demografici e le caratteristiche dell'ambiente virtuale in cui ciascun soggetto ha navigato.

All'ambiente virtuale sono associati due attributi: *textures* e *avatar*. L'attributo *textures* indica la qualità dell'ambiente virtuale ed è espressa in termini di numero di textures utilizzate per conferire il materiale alle varie strutture virtuali.

E' rappresentato da una variabile binaria che assume due valori, 1 o 2:

- ❖ 1: se la qualità dell'ambiente è bassa (poche textures);
- ❖ 2: se la qualità dell'ambiente è alta (tante textures).

L'attributo *avatar* indica la qualità degli avatars che popolano l'ambiente in cui il soggetto ha navigato. E' rappresentato da una variabile binaria che assume due valori, 1 o 2:

- ❖ 1: se la qualità degli avatar è bassa (*cartoon-like avatar*);
- ❖ 2: se la qualità degli avatar è alta (*photo realistic avatar*)

Nella tabella A-1 sono indicate, per ciascun soggetto sottoposto all'esperimento, le caratteristiche dell'ambiente virtuale.

ID	Textures	Avatar
11111	1	1
11121	1	2
11211	2	1
11221	2	2
12111	1	1
12121	1	2
12211	2	1
12221	2	2
21111	1	1
21121	1	2
21211	2	1
21221	2	2
22111	1	1
22121	1	2
22211	2	1
22221	2	2
31111	1	1
31121	1	2
31211	2	1
31221	2	2
32111	1	1
32121	1	2
32211	2	1
32221	2	2
41111	1	1
41121	1	2
41211	2	1
41221	2	2
42111	1	1
42121	1	2
42211	2	1
42221	2	2
51111	1	1
51121	1	2
51211	2	1
51221	2	2
52111	1	1
52121	1	2
52211	2	1
52221	2	2

Tabella A-1 Caratteristiche dell'ambiente virtuale per ciascun soggetto

Le altre informazioni che sono risultate utili nel processo di classificazione sono rappresentate dai dati demografici dei partecipanti.

Quelle più interessanti sono:

- ❖ *Age*: età del partecipante;
- ❖ *Gender*: sesso del partecipante. Indicato con una variabile binaria che ha due valori possibili 1 o 2:
 - 1: maschio;
 - 2: femmina.
- ❖ *Occupy*: indica l'attività lavorativa del partecipante. La variabile che tiene traccia di questa informazione ha otto valori possibili:
 - 1: undergrate student;
 - 2: masters student;
 - 3: PhD student;
 - 4: Research Assistant / Fellow;
 - 5: Staff – systems, technical;
 - 6: Faculty;
 - 7: Administrative Staff;
 - Other.
- ❖ *Medicine*: indica se il partecipante assume abitualmente medicinali. Indicato con una variabile binaria che ha due valori possibili, 1 o 2:
 - 1: true;
 - 2: false.
- ❖ *Language*: indica il grado di conoscenza dell'inglese da parte del partecipante. Questa informazione viene indicata con una variabile che ha tre valori possibili:
 - 1: basic;
 - 2: proficient;
 - 3: fluent.

❖ *Alcohol*: indica se il soggetto ha assunto sostanze alcoliche nelle sei ore precedenti all'esperimento. E' rappresentata con una variabile binaria che ha due valori possibili, 1 o 2:

- 1: true;
- 2: false.

Quest'ultima informazione è l'unica che non è stata considerata nella nostra analisi perché poco significativa: tutti avevano assunto sostanze alcoliche prima di essere sottoposti all'esperimento.

Le informazioni relative a questi parametri sono riportate nella tabella A-2.

Sono state considerate anche informazioni sui partecipanti relative al livello di conoscenze informatiche, di realtà virtuale ed esperienze con i videogames.

In particolare, sono stati considerati i seguenti attributi:

- ❖ *Literate*: indica il livello di conoscenza del computer. Questa informazione viene indicata con un valore compreso tra 1 e 7. Valori vicini ad 1 indicano poca conoscenza mentre valori prossimi a 7 indicano molta conoscenza.
- ❖ *Programming*: indica il livello di conoscenza di programmazione. Anche questa informazione viene indicato con un valore compreso tra 1 e 7.
- ❖ *VReality*: indica il numero di esperienze virtuali vissute dal partecipante.
- ❖ *Games*: indica il numero di volte in cui il partecipante ha giocato con un videogames nell'ultimo anno. Rappresentata da una variabile che ha sette valori possibili:
 - 1: never;
 - 2: 1-5;
 - 3: 6-10;
 - 4: 11-15;
 - 5: 16-20;
 - 6: 21-25;
 - 7: >25.

❖ *GameTimes*: indica il numero di ore trascorse dal soggetto con un videogames nell'ultima settimana. Indicato con un valore compreso tra 1 e 7, dove:

- 1: 0;
- 2: < 1 ora;
- 3: 1-3;
- 4: 3 - 5;
- 5: 5 - 7;
- 6: 7 – 9;
- 7: > 9.

In tabella A-3 sono riportate queste informazioni per ciascun soggetto.

ID	Age	Gender	Occupy	Medicine	Language	Alcohol
11111	26	1	2	2	3	2
11121	34	1	8	2	3	2
11211	30	1	2	2	3	2
11221	24	1	2	2	3	2
12111	24	2	2	2	3	2
12121	20	2	1	2	3	2
12211	35	2	8	2	3	2
12221	27	2	8	1	3	2
21111	43	1	3	2	3	2
21121	24	1	1	2	3	2
21211	25	1	3	2	3	2
21221	24	1	2	1	3	2
22111	29	2	8	2	3	2
22121	34	2	2	1	3	2
22211	30	2	2	2	3	2
22221	19	2	1	2	3	2
31111	23	1	2	2	3	2
31121	18	1	1	2	3	2
31211	25	1	2	1	3	2
31221	26	1	2	2	3	2
32111	40	2	2	2	3	2
32121	32	2	3	2	3	2
32211	28	2	2	2	3	2
32221	18	2	1	2	3	2
41111	25	1	8	2	3	2
41121	28	1	2	2	3	2
41211	21	1	1	2	2	2
41221	22	1	2	2	3	2
42111	21	2	1	2	3	2
42121	19	2	1	1	3	2
42211	41	2	6	2	3	2
42221	20	2	1	2	3	2
51111	28	1	2	2	3	2
51121	19	1	1	2	3	2
51211	19	1	1	2	3	2
51221	23	1	2	2	3	2
52111	27	2	3	2	3	2
52121	20	2	1	2	2	2
52211	18	2	1	1	3	2
52221	21	2	1	2	3	2

Tabella A-2 Dati demografici dei partecipanti all'esperimento

ID	Literate	Programming	VReality	Games	Gametimes
11111	5	2	1	5	3
11121	3	1	4	1	1
11211	6	3	3	2	1
11221	5	2	2	7	3
12111	6	4	1	1	1
12121	3	1	1	2	1
12211	3	1	1	1	1
12221	4	2	3	2	1
21111	3	1	1	1	1
21121	5	2	1	7	7
21211	5	5	1	2	1
21221	4	1	1	4	1
22111	7	5	5	2	1
22121	3	1	4	2	1
22211	4	1	1	1	1
22221	4	2	3	6	2
31111	6	4	2	7	3
31121	4	1	1	7	5
31211	6	4	2	1	1
31221	6	2	1	7	4
32111	5	1	2	1	1
32121	4	1	1	3	1
32211	4	1	1	2	2
32221	3	5	3	2	2
41111	5	2	1	7	4
41121	6	3	2	2	1
41211	4	1	2	2	1
41221	6	5	5	7	3
42111	5	1	2	4	2
42121	3	1	1	4	2
42211	7	1	1	1	1
42221	4	1	1	1	1
51111	5	1	3	2	1
51121	3	1	1	2	2
51211	4	1	2	5	2
51221	6	5	2	7	6
52111	5	1	1	2	2
52121	5	2	1	4	2
52211	5	1	1	6	3
52221	5	1	1	1	1

Tabella A-3 Conoscenze informatiche e di realtà virtual di ciascun partecipante

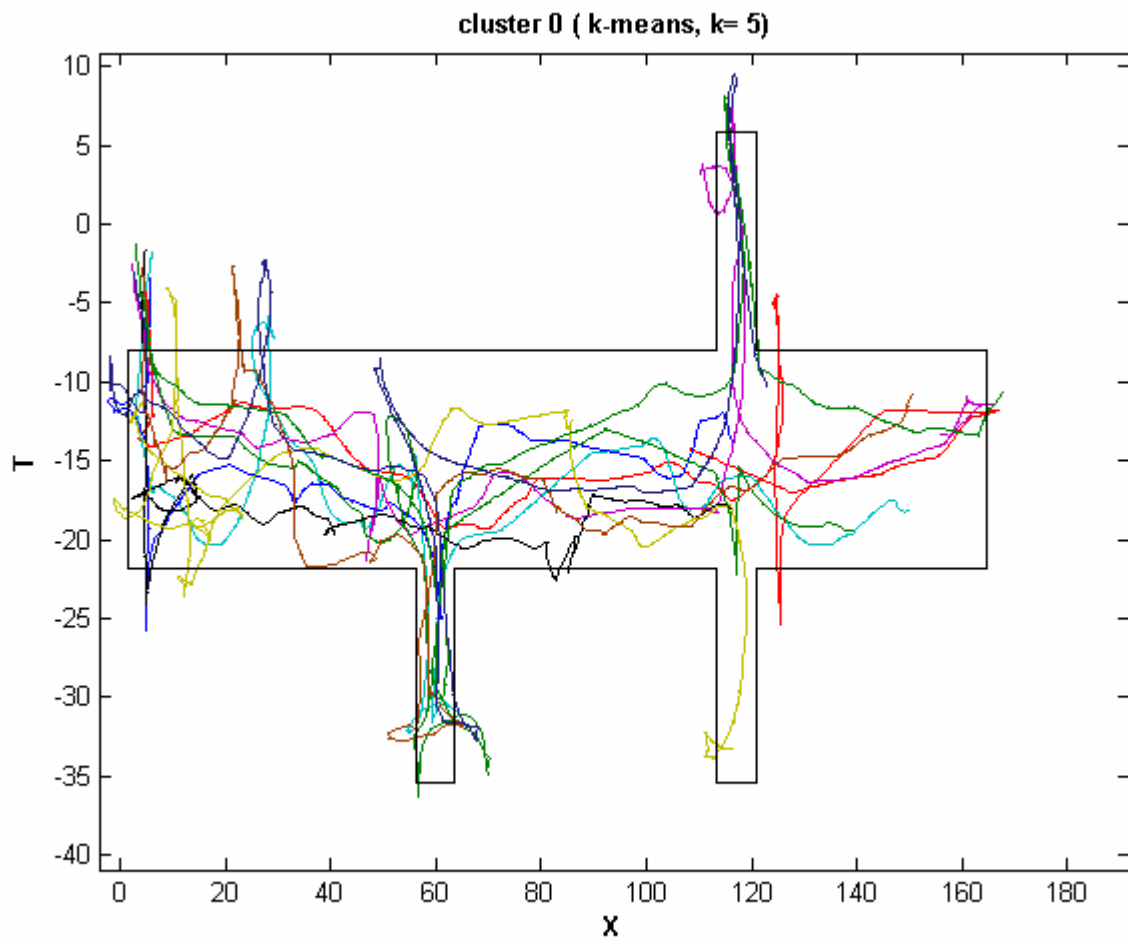
Appendice B

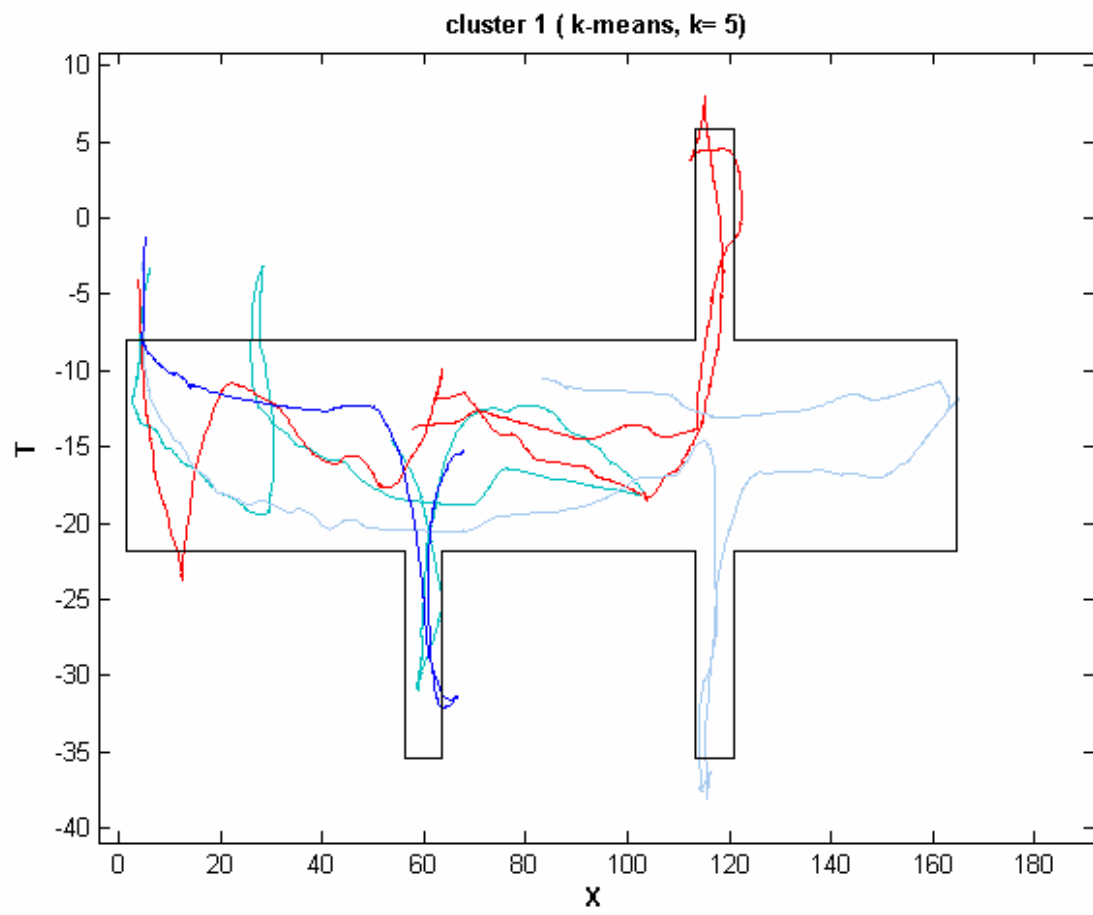
La prima fase dell'analisi descritta nel paragrafo 6.4 ci ha permesso di suddividere le traiettorie in clusters in maniera che quelle appartenenti allo stesso clusters siano molto simili e risultino molto diverse da qualunque traiettoria appartenente ad altri clusters. L'individuazione dei clusters è avvenuta considerando due diversi algoritmi: uno di tipo k-means e l'altro di tipo gerarchico. Dopo l'esecuzione dei due algoritmi per vari k si è riscontrato che in entrambi i casi si raggiunge una stabilità nei clusters con $k=5$ e $k=6$. In tabella B-1 è indicato per ciascuna traiettoria il cluster di appartenenza a seconda dell'algoritmo e a seconda del numero di clusters.

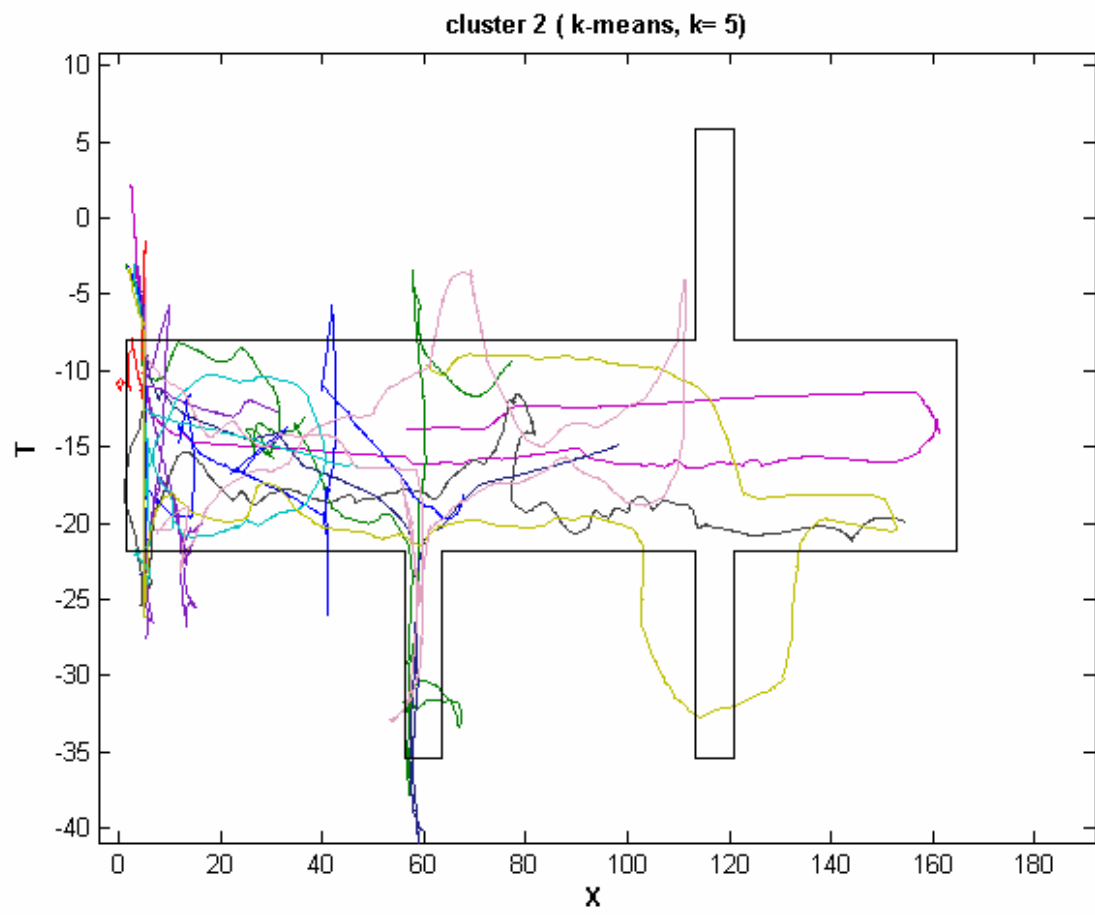
Nelle pagine successive è fornita una rappresentazione dei clusters determinati con l'algoritmo k-means e con il gerarchico, sia per $k=5$ che per $k=6$.

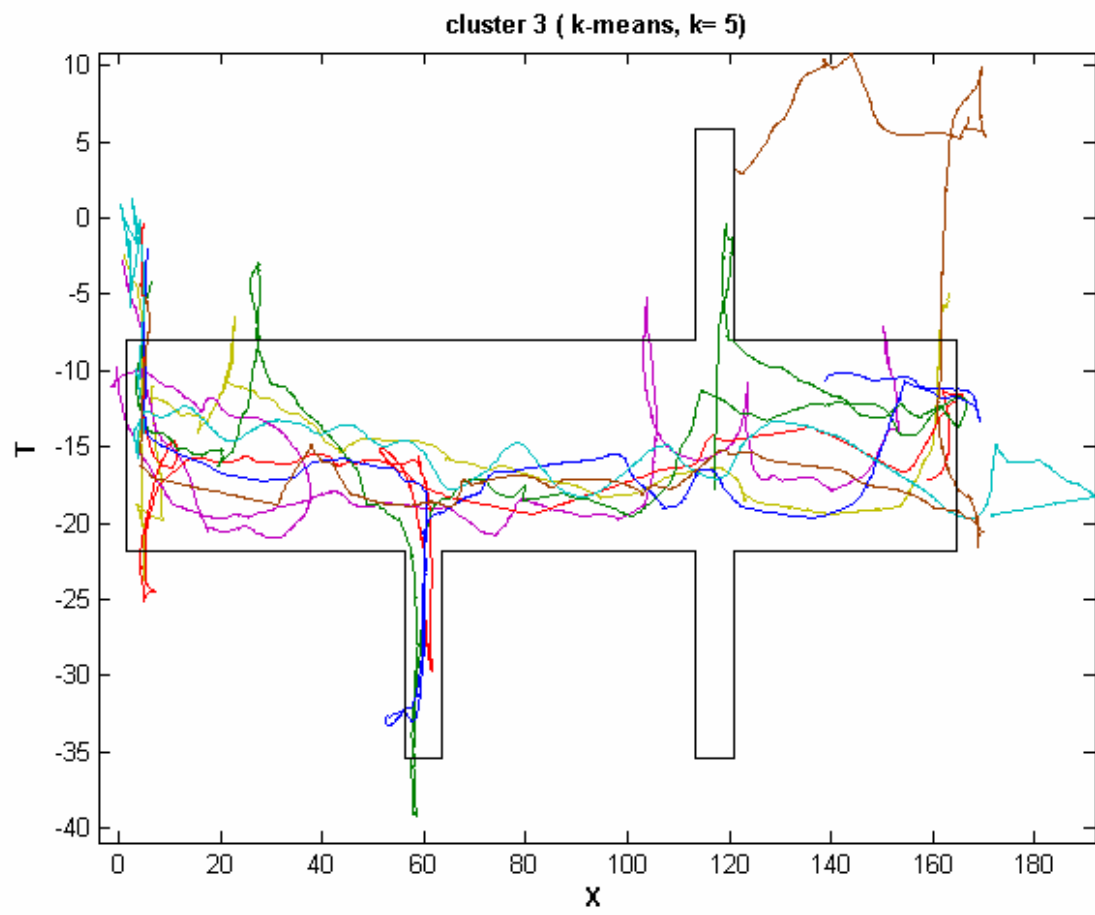
	<i>k-means</i>		<i>Gerarchico</i>	
<i>Ide_traj</i>	5	6	5	6
11111	4	4	3	4
11121	4	3	3	4
11211	0	0	3	4
11221	0	3	2	3
12111	2	4	4	5
12121	4	4	4	5
12211	2	1	4	5
12221	4	4	4	5
21111	2	4	4	5
21121	0	0	3	4
21211	1	1	1	2
21221	0	3	2	3
22111	3	5	2	3
22121	2	1	4	1
22211	2	4	4	5
22221	0	0	2	3
31111	4	0	3	4
31121	0	0	3	4
31211	3	5	2	3
31221	0	0	3	4
32111	2	4	4	5
32121	4	4	3	4
32211	2	1	4	1
32221	1	1	1	2
41111	3	5	2	3
41121	4	4	3	4
41211	4	4	4	5
41221	0	3	3	4
42111	0	3	2	3
42121	1	2	1	2
42211	2	1	4	1
42221	1	1	4	5
51111	2	4	4	5
51121	4	4	3	4
51211	2	1	4	1
51221	3	0	0	0
52111	3	0	2	3
52121	3	5	2	3
52211	0	0	3	4
52221	3	2	0	0

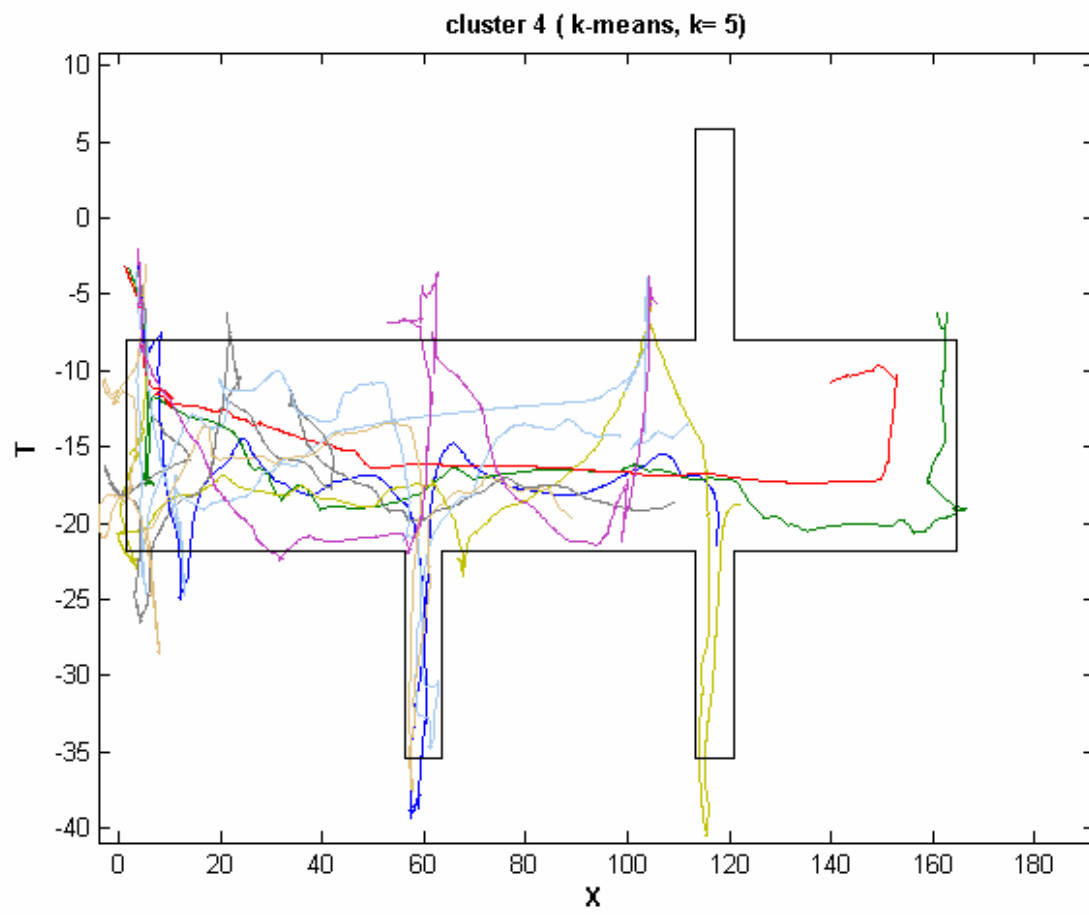
Tabella B-1: Suddivisione delle traiettorie in clusters

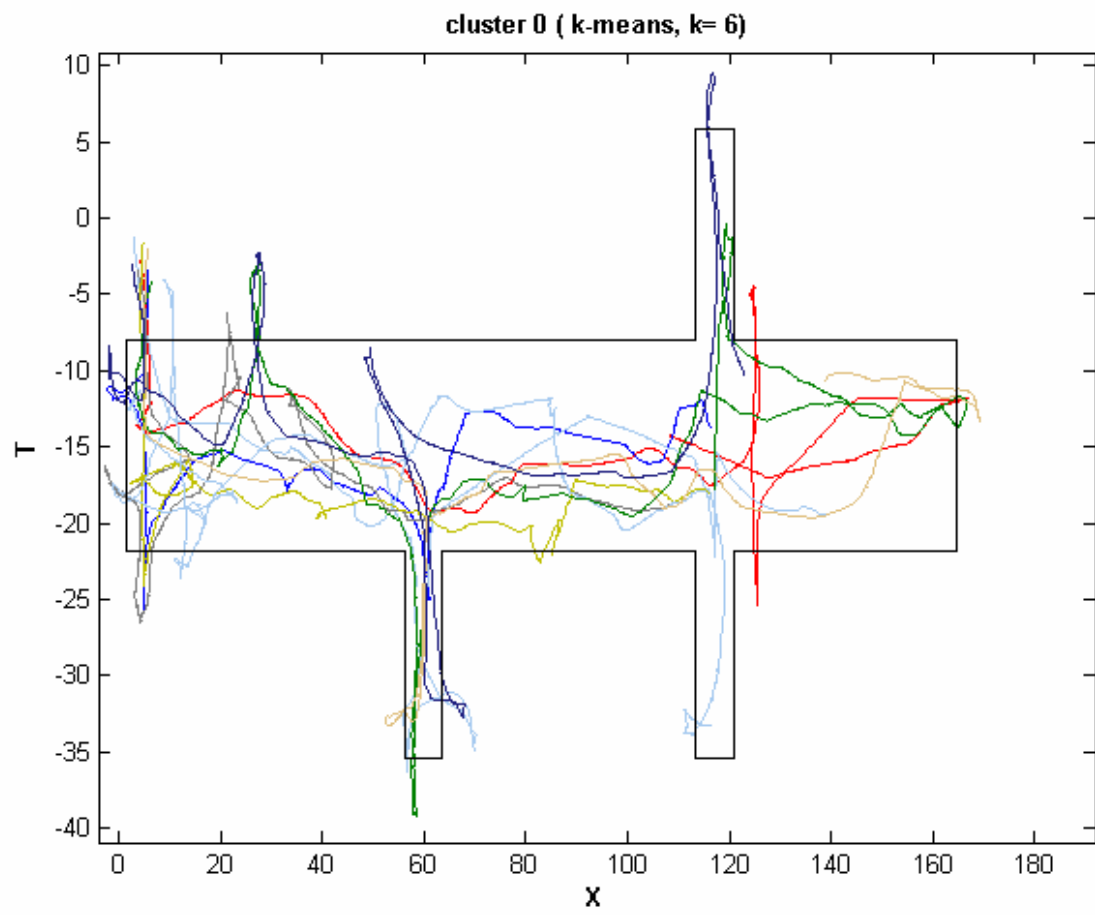


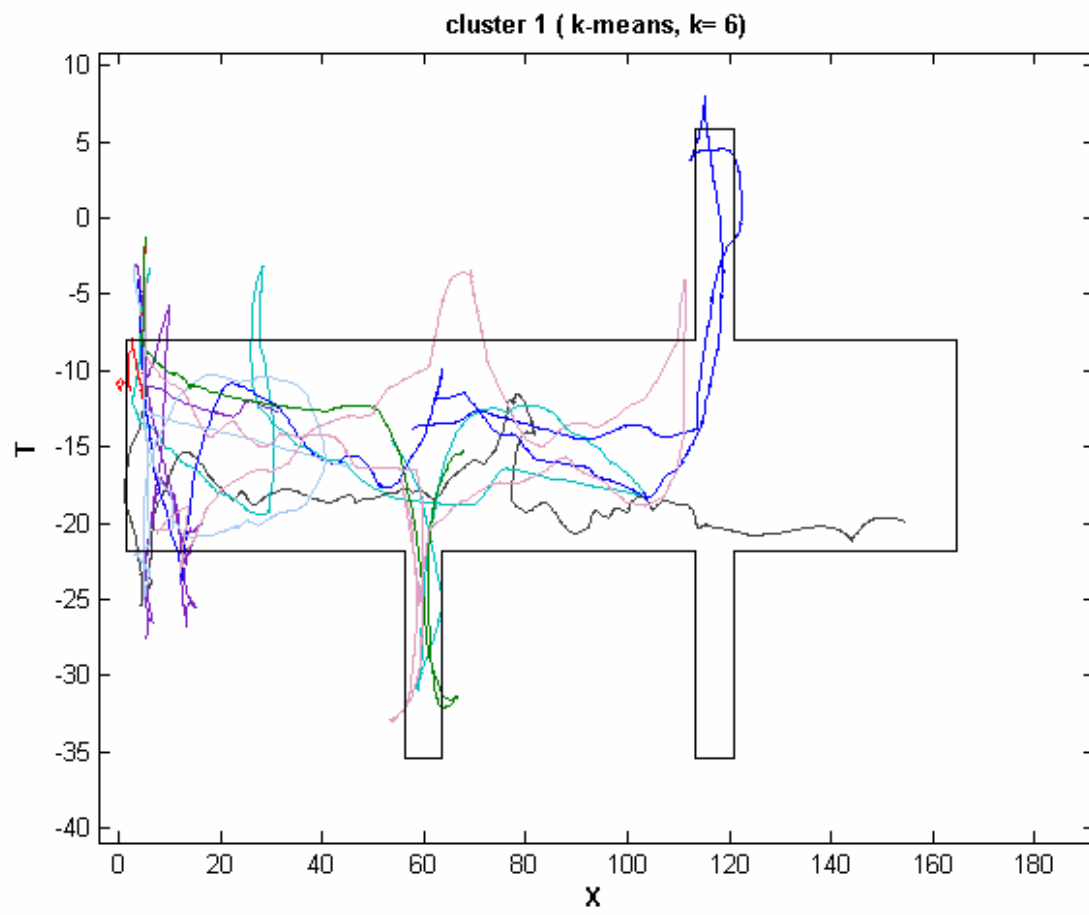


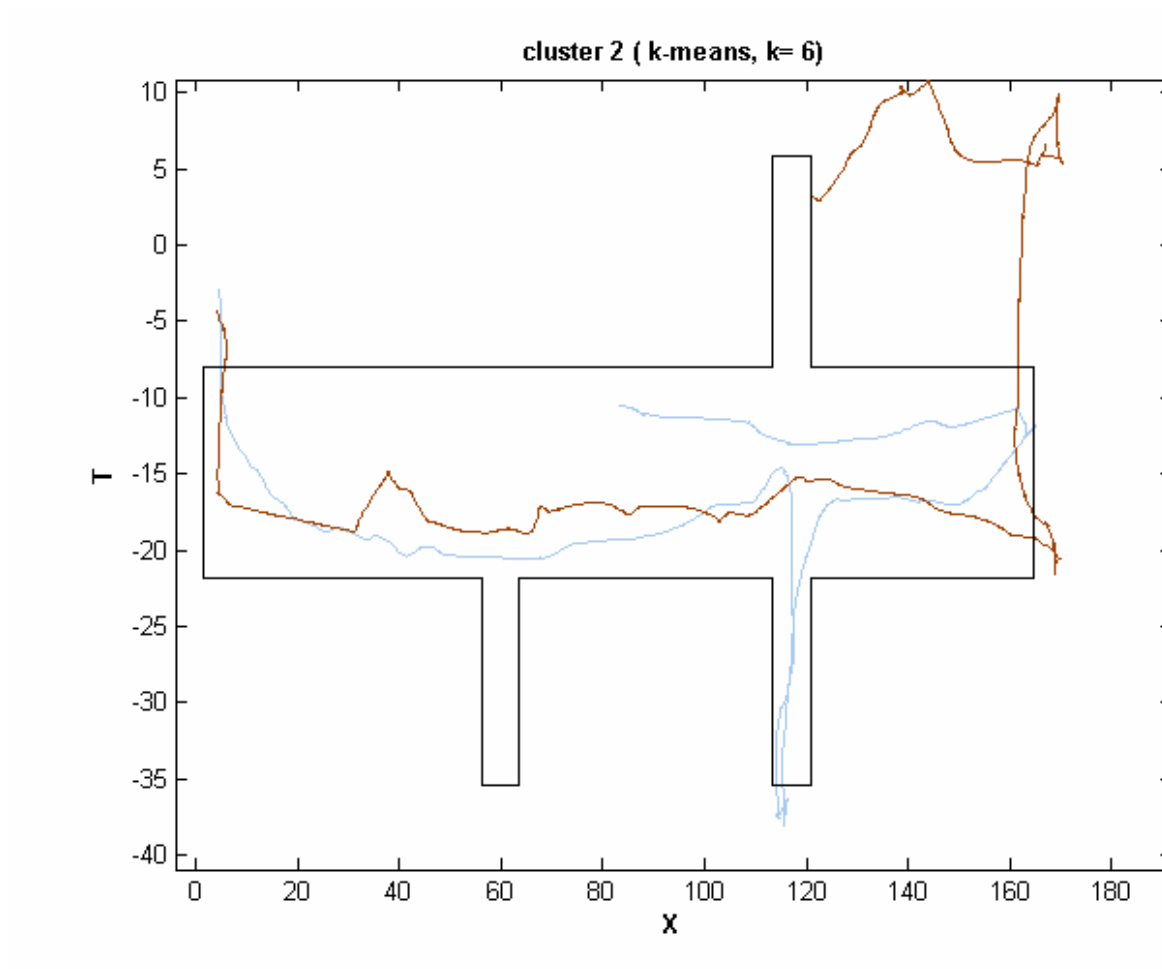


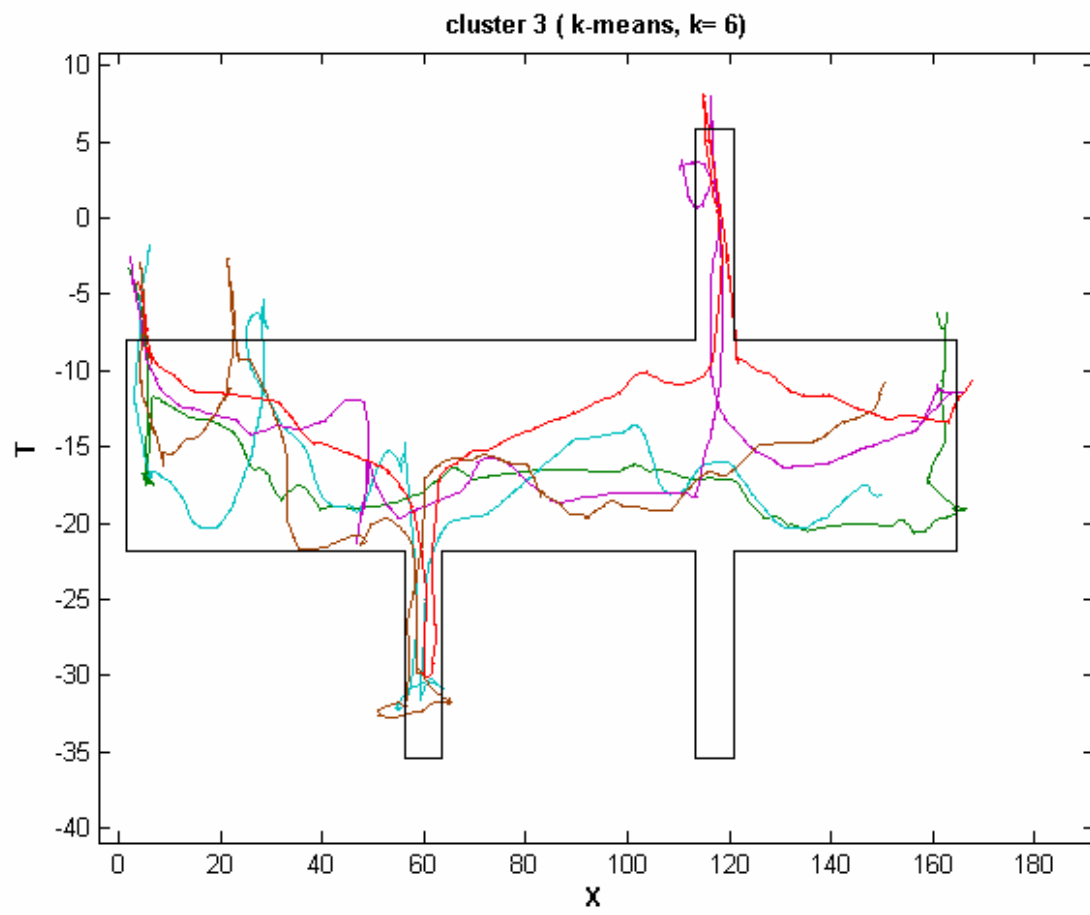


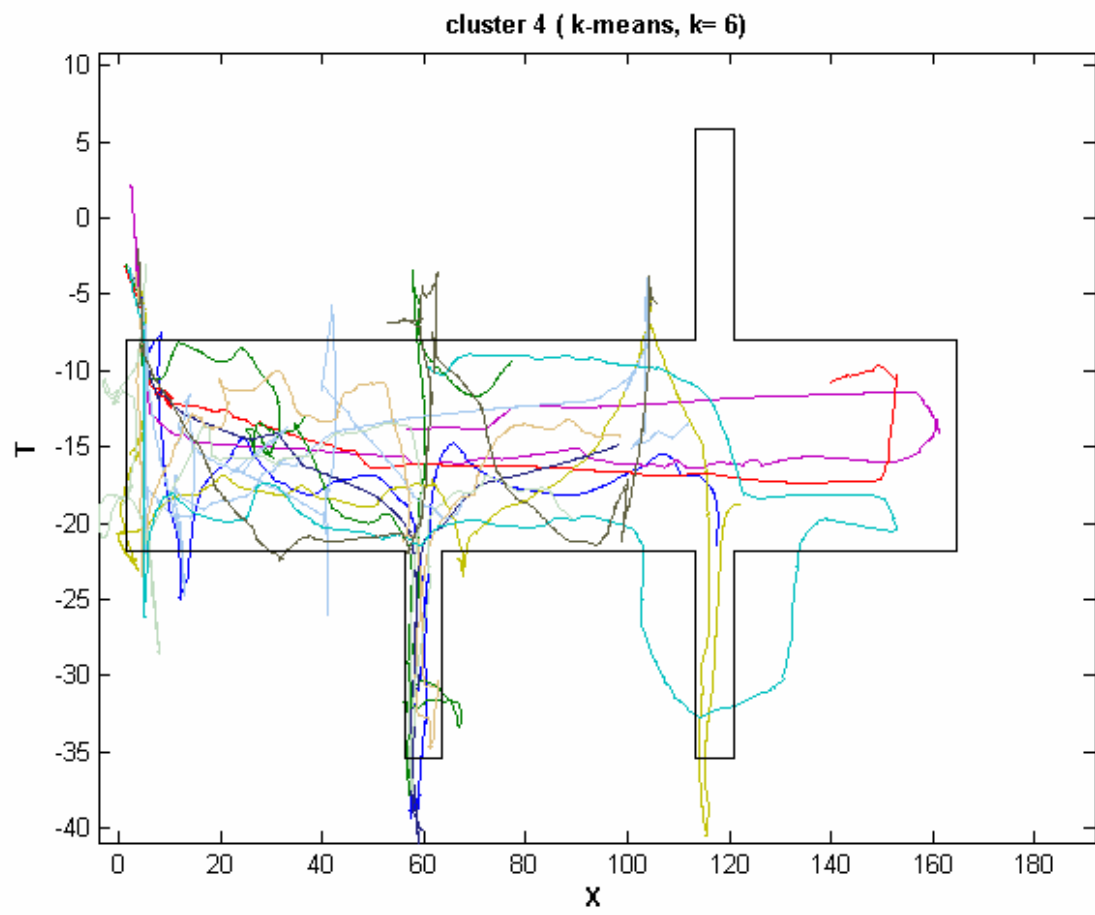


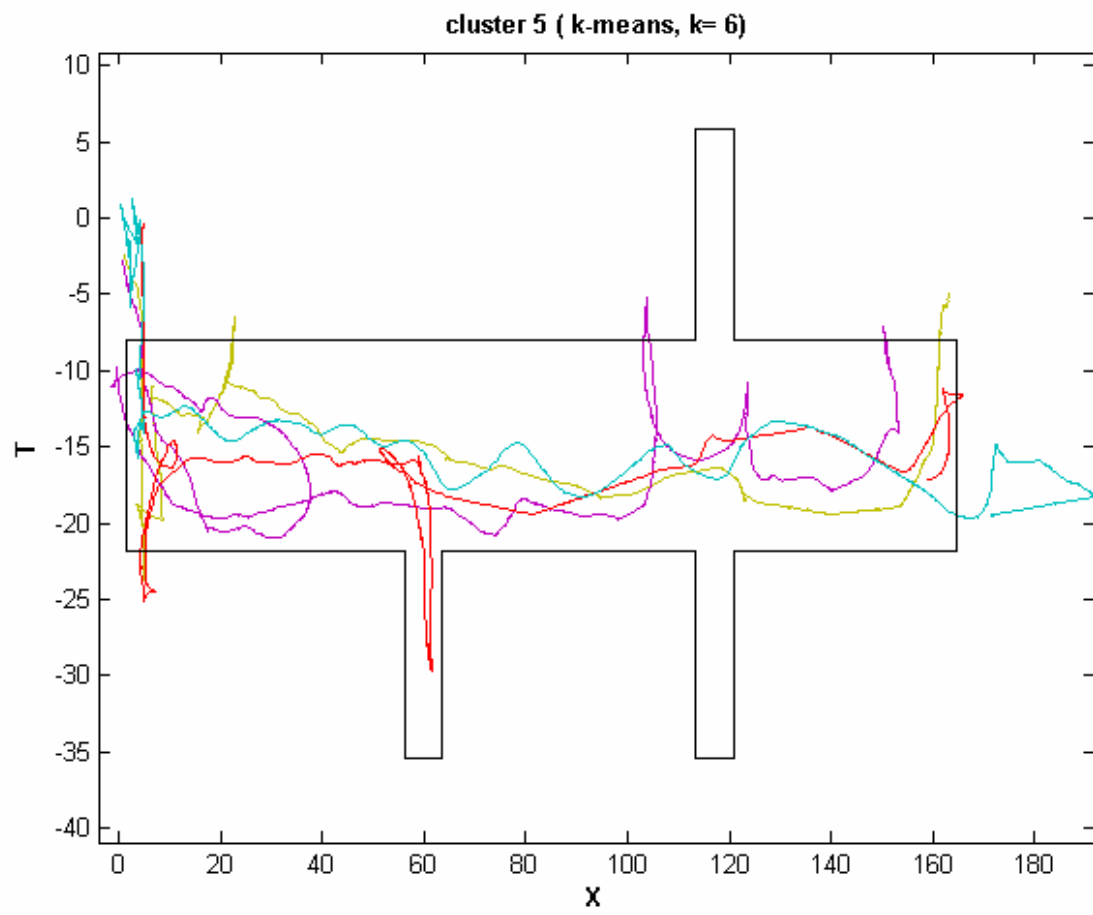


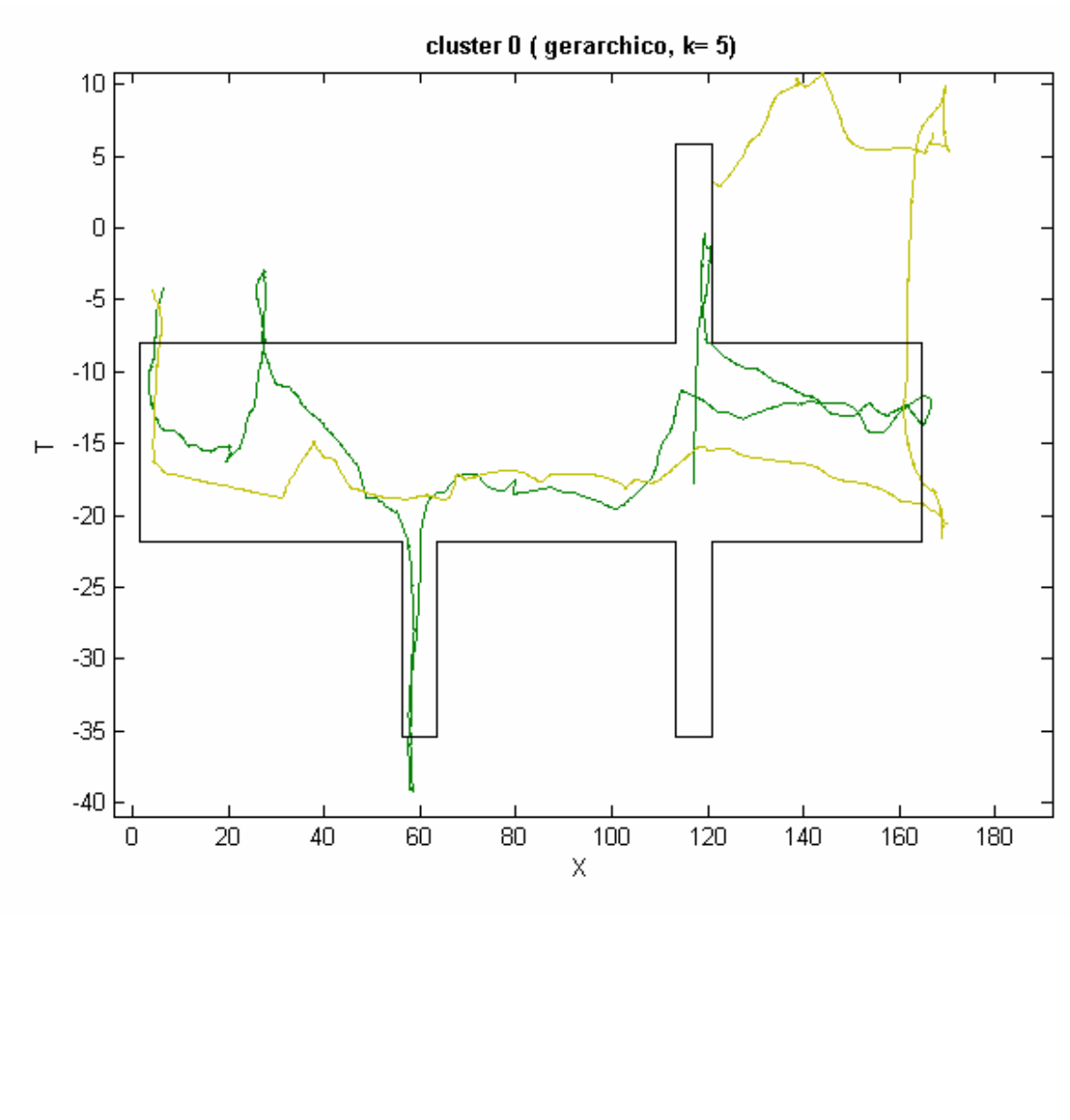


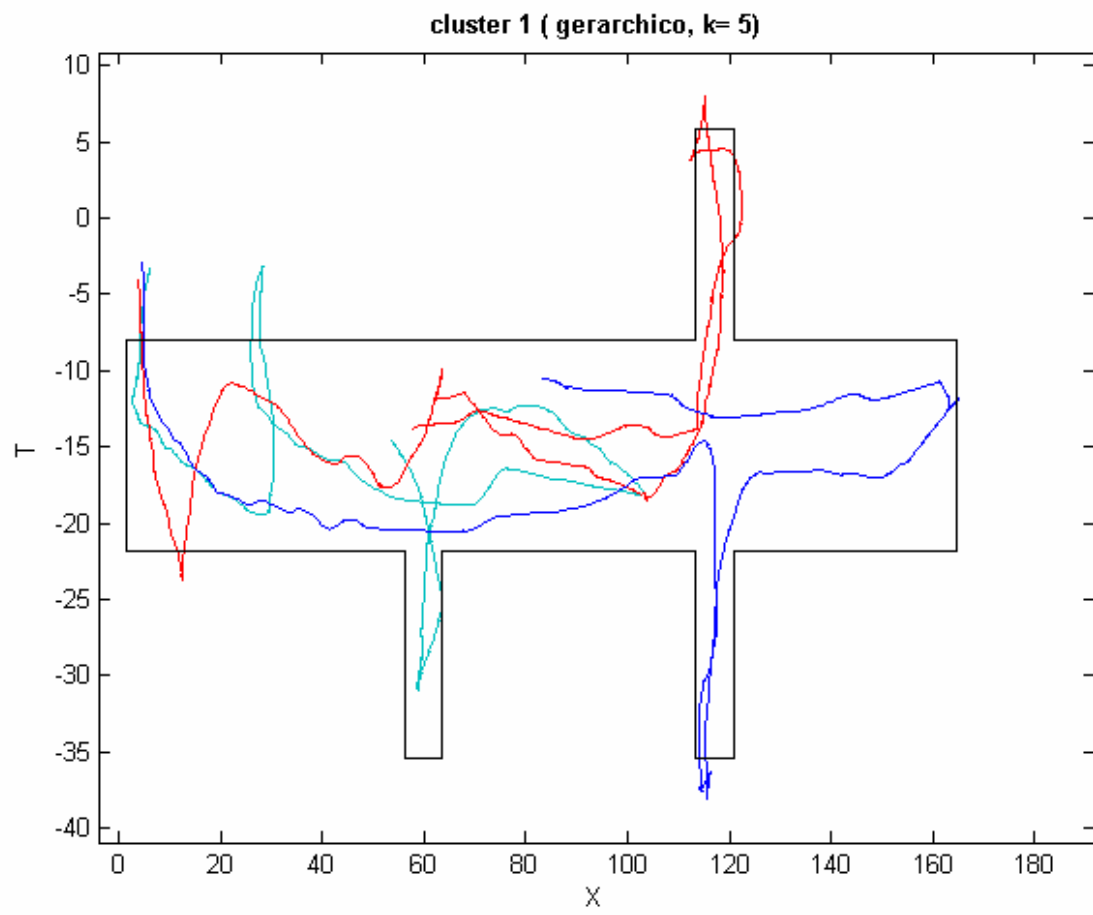


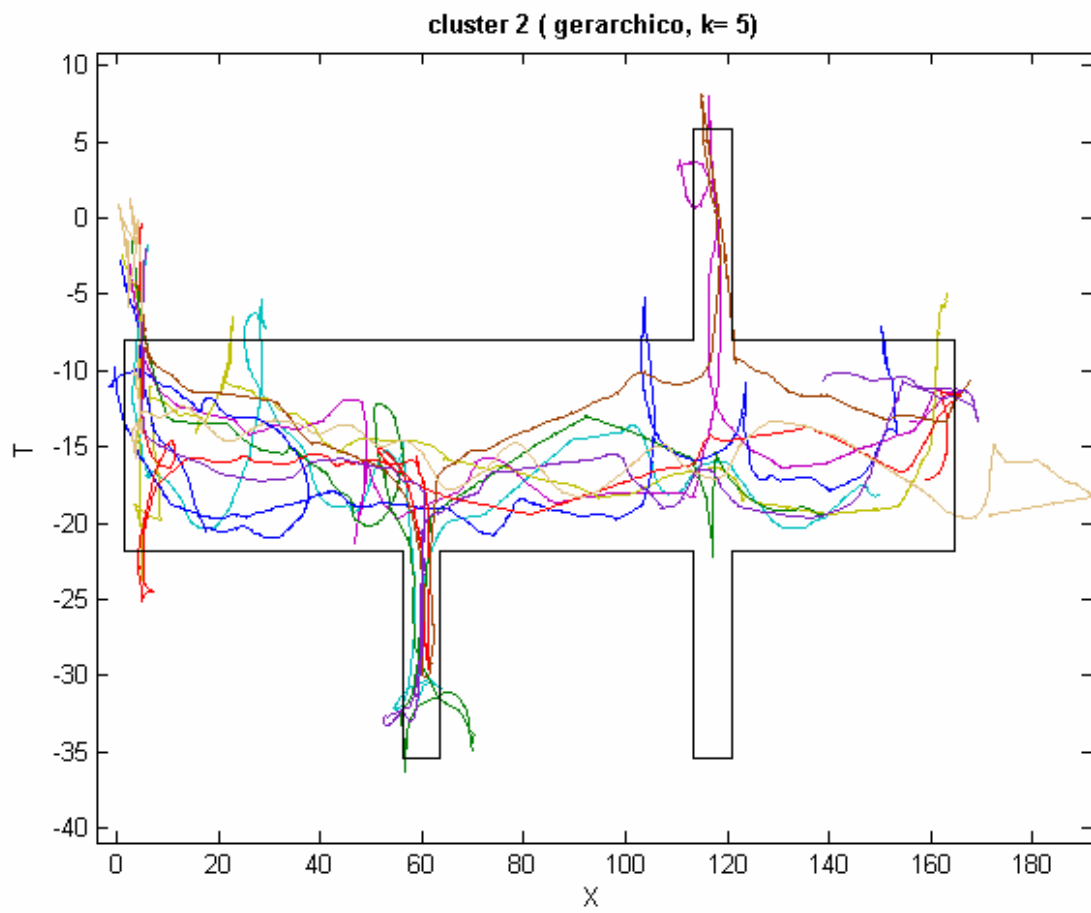


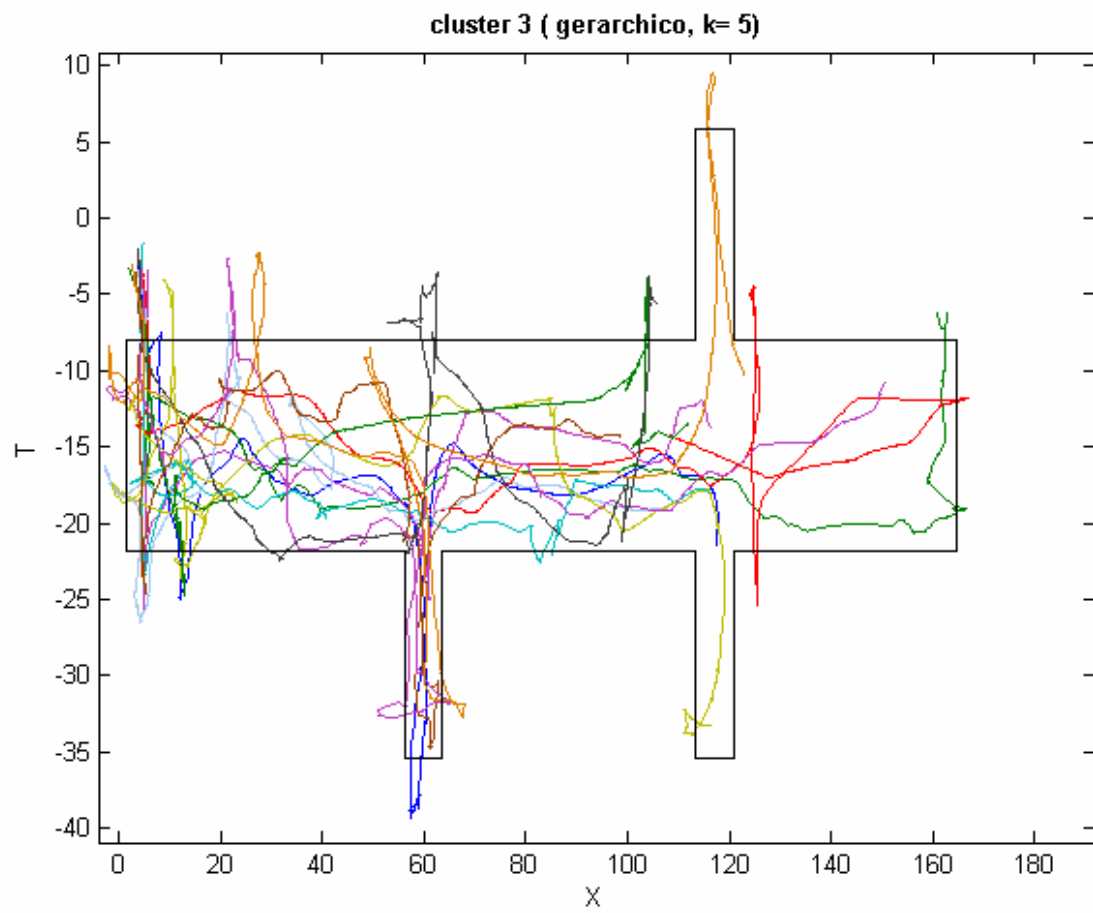


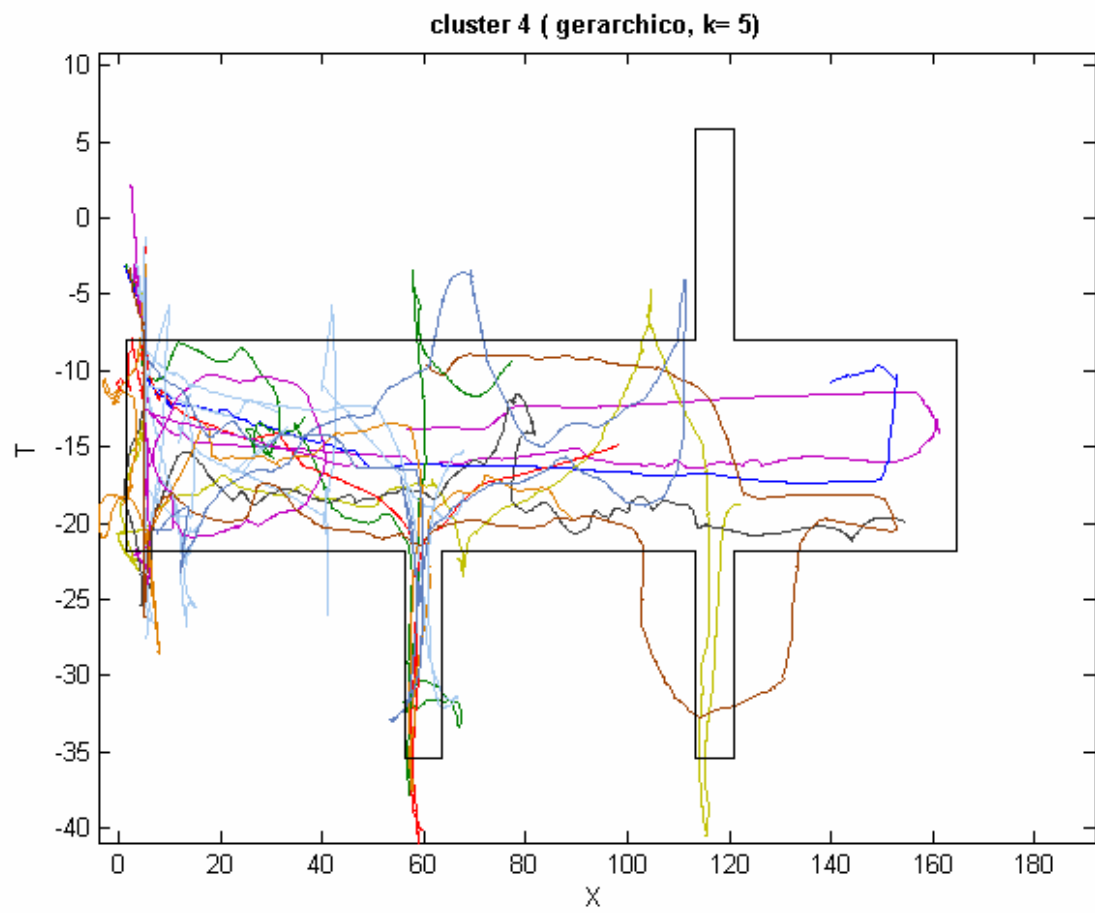


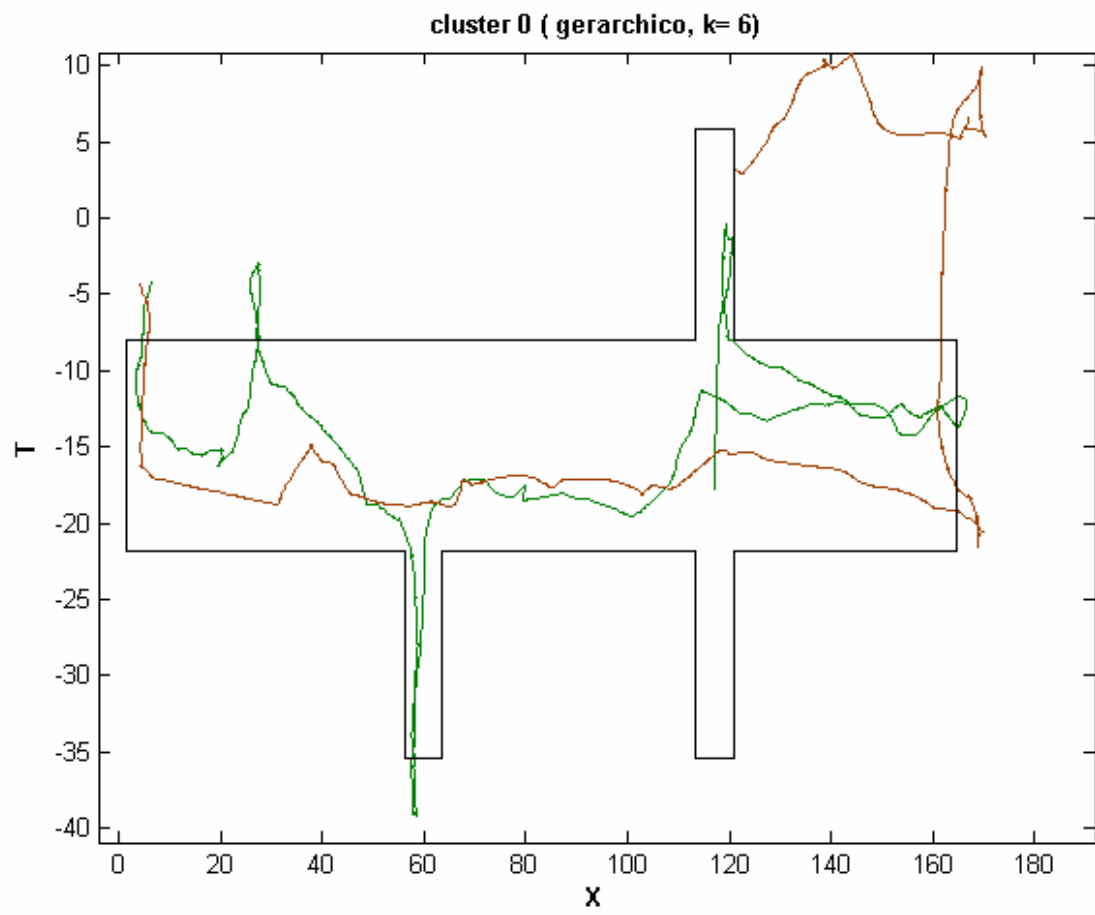


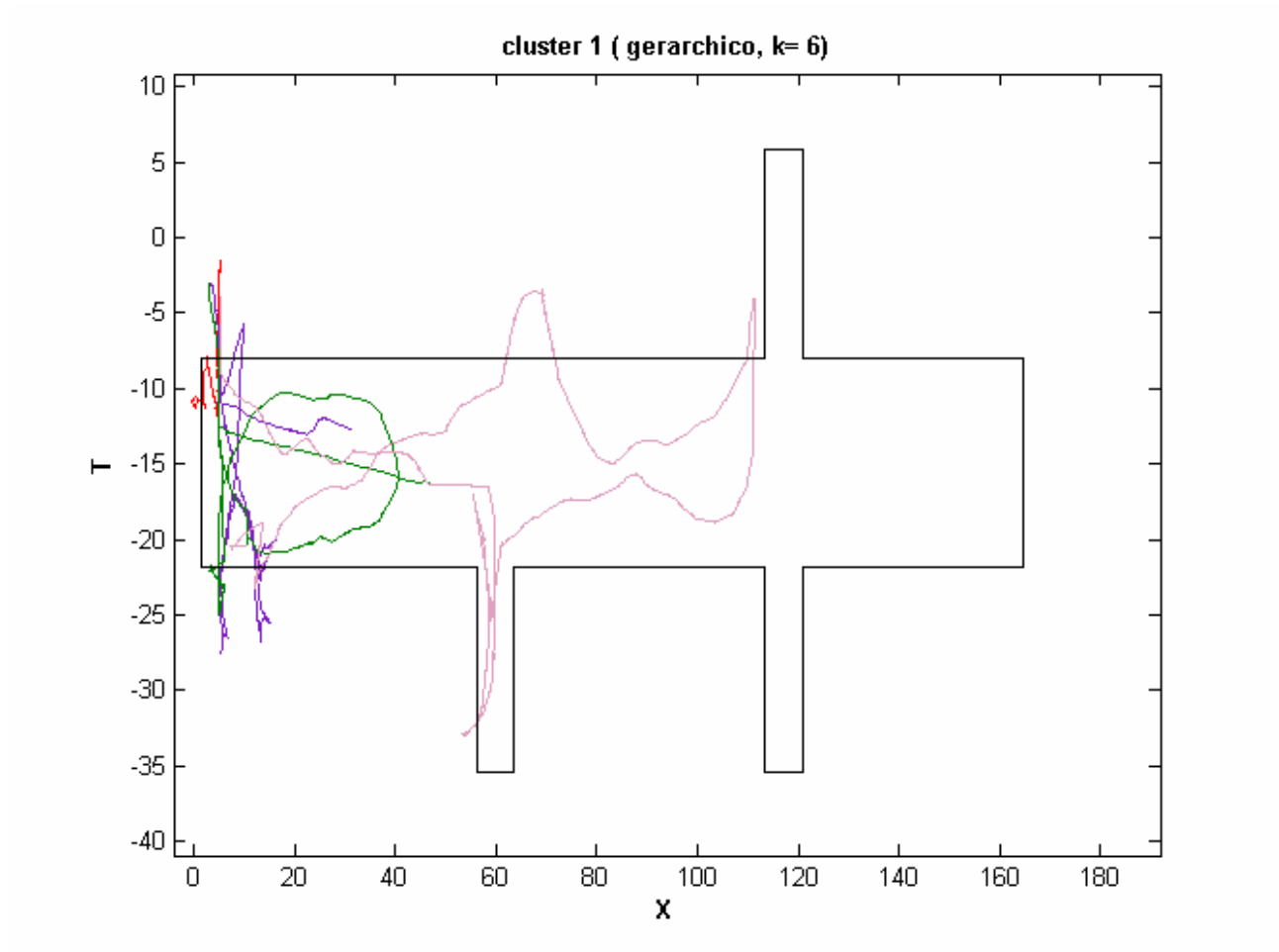


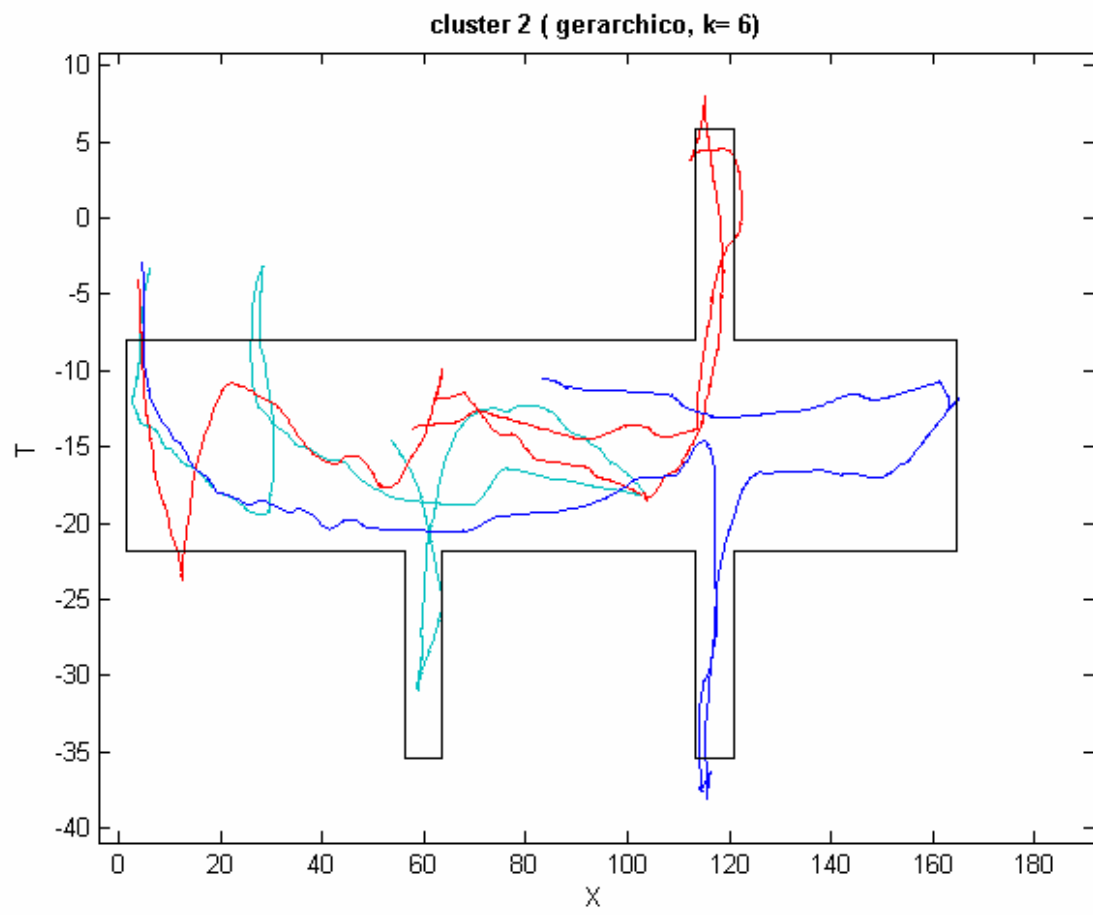


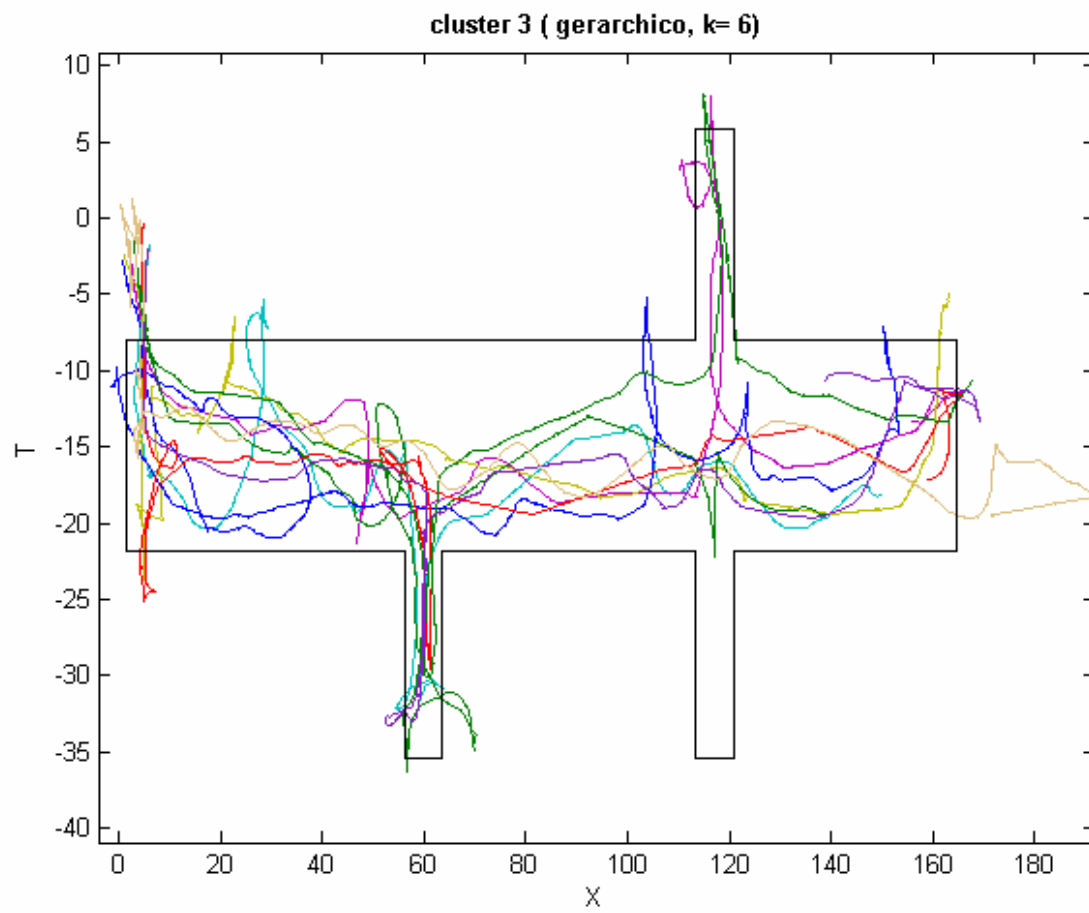


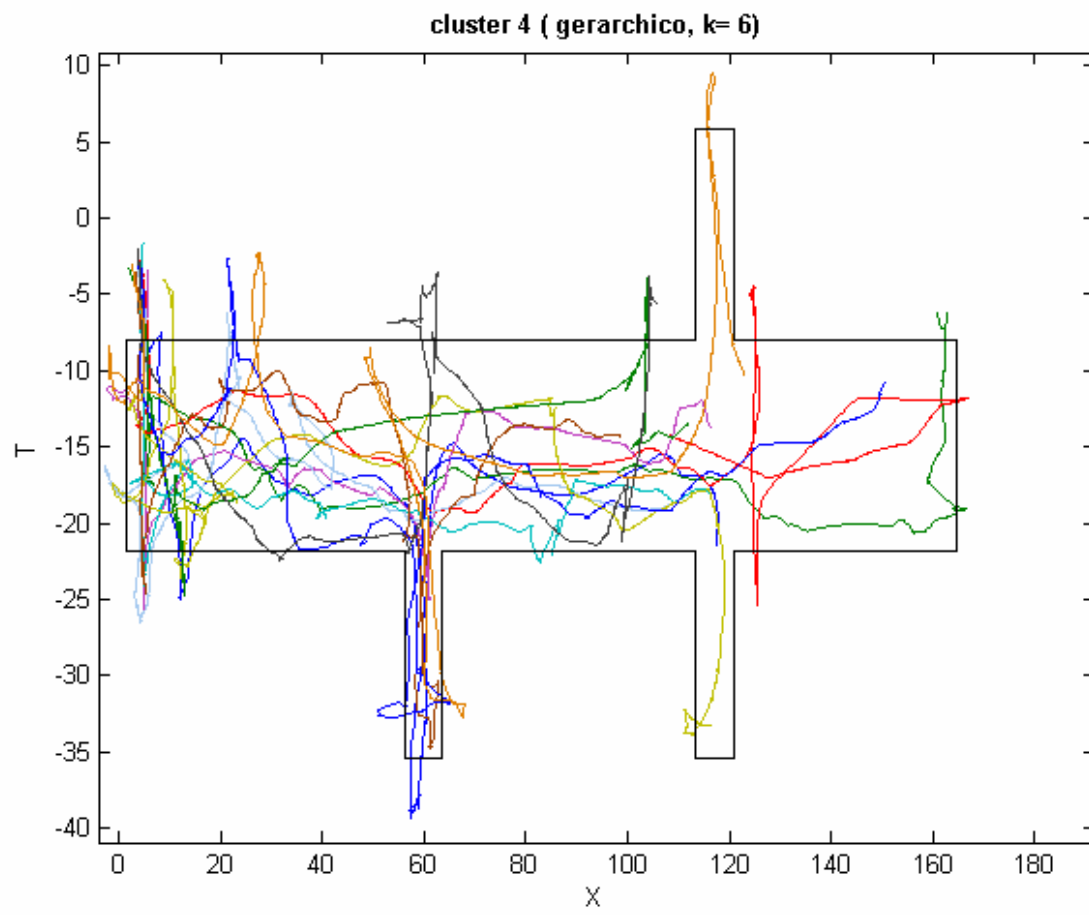


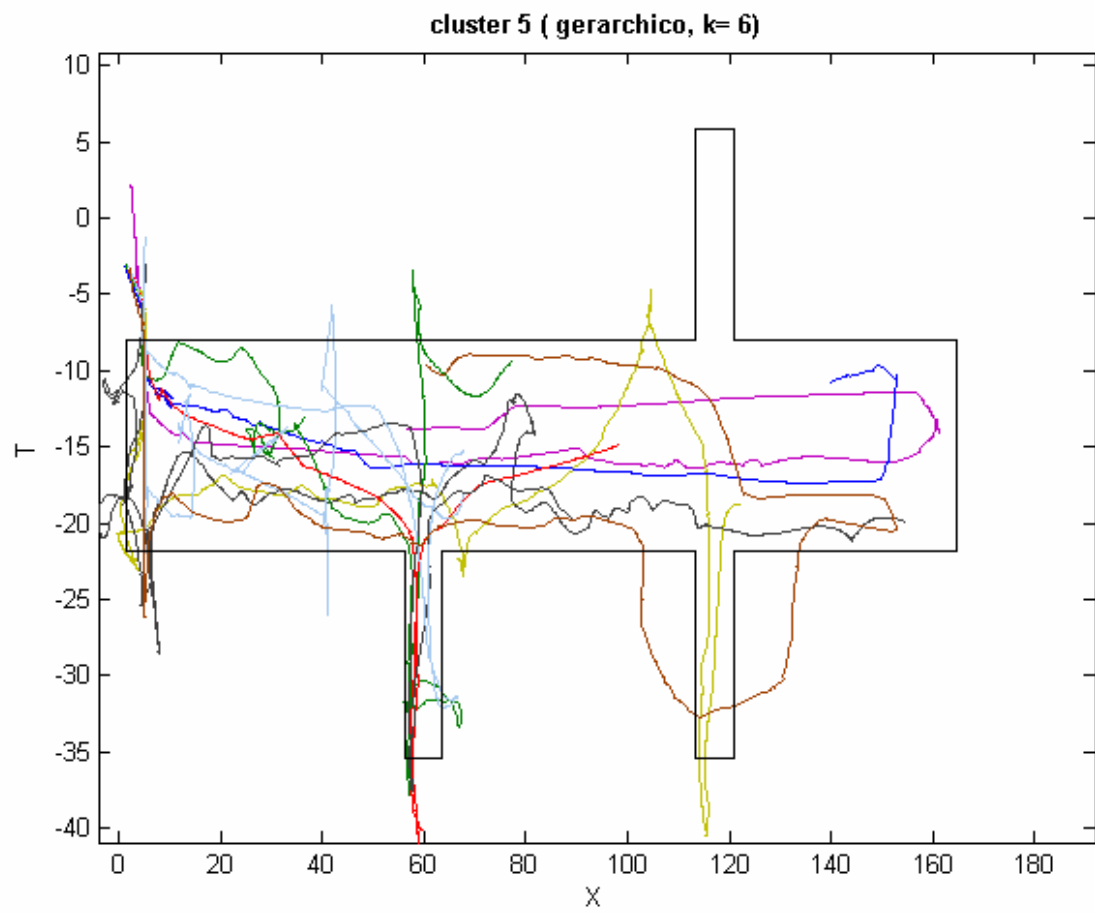












Processo di classificazione: regole di decisione

La fase di classificazione descritta nel paragrafo 6.7 è stata eseguita per passi successivi considerando ad ogni iterazione solo alcuni attributi.

Ogni classificazione ha prodotto un albero di decisione che ci ha permesso di estrarre le regole che vengono riportate di seguito. L'output di ciascun classificazione è indicato con una lettera dell'alfabeto. Nella tabella B-2 sono indicati per ciascuna classificazione gli attributi considerati.

A)

➤ Caso: Clusters determinati con k-means e k= 5

- Se (numero_BIPs = 0) allora numero_cluster= 0 (5.0/4.1)
- Se (numero_BIPs = 1) allora numero_cluster= 0 (3.0/2.8)
- Se (numero_BIPs = 2) allora numero_cluster= 4 (8.0/6.3)
- Se (numero_BIPs = 3) allora numero_cluster= 0 (7.0/5.3)
- Se (numero_BIPs = 4) allora numero_cluster= 0 (4.0/3.1)
- Se (numero_BIPs = 5) allora numero_cluster= 3 (6.0/4.3)
- Se (numero_BIPs = 6) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs = 8) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs = 10) allora numero_cluster= 0 (1.0/0.8)
- Se (numero_BIPs = 11) allora numero_cluster= 2 (2.0/1.8)
- Se (numero_BIPs = 16) allora numero_cluster= 3 (1.0/0.8)

➤ Caso: Clusters determinati con k-means e k= 6

- numero_cluster= 4 (40.0/30.4)

➤ Caso: Clusters determinati con gerarchico e k= 5

- numero_cluster= 4 (40.0/28.5)

➤ Caso: Clusters determinati con gerarchico e k= 6

- Se (numero_BIPs = 0) allora numero_cluster= 5 (5.0/4.1)
- Se (numero_BIPs = 1) allora numero_cluster= 0 (3.0/2.8)
- Se (numero_BIPs = 2) allora numero_cluster= 4 (8.0/6.3)
- Se (numero_BIPs = 3) allora numero_cluster= 4 (7.0/4.4)

- Se (numero_BIPs = 4) allora numero_cluster= 3 (4.0/3.1)
- Se (numero_BIPs = 5) allora numero_cluster= 3 (6.0/4.3)
- Se (numero_BIPs = 6) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs = 8) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs = 10) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs = 11) allora numero_cluster= 5 (2.0/1.0)
- Se (numero_BIPs = 16) allora numero_cluster= 0 (1.0/0.8)

B)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (mediaGSR > 0.102) allora numero_cluster= 4 (6.0/2.3)
- Se ((mediaGSR <= 0.102) & (mediaResp <= -147070) & (avatar= cartoon)) allora numero_cluster= 2 (9.0/5.5)
- Se ((mediaGSR <= 0.102) & (mediaResp <= -147070) & (avatar= realistico) & (sesso= femmina)) allora numero_cluster= 4 (4.0/3.1)
- Se ((mediaGSR <= 0.102) & (mediaResp<=-147070) & (avatar= realistico) & (sesso= maschio)) allora numero_cluster= 0 (6.0/2.3)
- Se ((mediaGSR <= 0.102) & (-147070 < mediaResp <= -71401) & (texture=alta)) allora numero_cluster= 0 (2.0/1.8)
- Se ((mediaGSR <= 0.102) & (texture= alta) & (-71401 < mediaResp <= -48896)) allora numero_cluster= 1 (2.0/1.0)
- Se ((mediaGSR <= 0.102) & (texture=alta) & (mediaResp > -48896)) allora numero_cluster= 3 (4.0/2.2)
- Se ((mediaGSR <= -0.008) & (mediaResp > -147070) & (texture= bassa)) allora numero_cluster= 2 (2.0/1.0)
- Se ((-0.008 < mediaGSR <= 0.102) & (mediaResp > -147070) & (texture= bassa)) allora numero_cluster= 3 (5.0/2.3)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((mediaResp <= -11852) & (texture= alta) & (sesso= femmina)) allora numero_cluster= 1 (9.0/5.5)
- Se ((mediaResp<= -11852) & (texture= alta) & (sesso= maschio) & (avatar= cartoon) & (mediaGSR <= 0.014)) allora numero_cluster= 1 (2.0/1.0)
- Se ((mediaResp<= -11852) & (texture= alta) & (sesso= maschio) & (avatar= cartoon) & (mediaGSR> 0.014)) allora numero_cluster= 0 (2.0/1.8)
- Se ((mediaResp<= -11852) & (texture= alta) & (sesso= maschio) & (avatar= realistico)) allora numero_cluster= 3 (3.0/1.1)
- Se ((mediaResp<= -11852) & (texture= bassa)) allora numero_cluster= 4 (16.0/9.8)

- Se ((mediaResp > -11852) & (sesso= femmina)) allora numero_cluster= 2 (3.0/2.8)
- Se((mediaResp > -11852) & (sesso= maschio) & (avatar= cartoon)) allora numero_cluster= 5 (3.0/2.1)
- Se ((mediaResp > -11852) & (sesso= maschio) & (avatar= realistico)) allora numero_cluster= 0 (2.0/1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((sesso= femmina) & (avatar= cartoon)) allora numero_cluster= 4 (10.0/5.6)
- Se ((sesso= femmina) & (avatar= realistico) & (mediaGSR <= 0.006)) allora numero_cluster= 4 (6.0/3.3)
- Se ((sesso= femmina) & (avatar= realistico) & (0.006 < mediaGSR <= 0.038)) allora numero_cluster= 2 (2.0/1.0)
- Se ((sesso= femmina) & (avatar= realistico) & (mediaGSR > 0.038)) allora numero_cluster= 1 (2.0/1.8)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaResp <= -270706)) allora numero_cluster= 1 (2.0/1.8)
- Se ((sesso= maschio) & (avatar= cartoon) & (-270706 < mediaResp <= -208974)) allora numero_cluster= 3 (2.0/1.0)
- Se ((sesso= maschio) & (avatar= cartoon) & (-208974 < mediaResp <= -11852)) allora numero_cluster= 4 (3.0/1.1)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaResp > -11852)) allora numero_cluster= 2 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= alta) & (mediaResp <= -208974)) allora numero_cluster= 2 (2.0/1.0)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= alta) & (mediaResp > -208974)) allora numero_cluster= 3 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= bassa)) allora numero_cluster= 3 (5.0/1.2)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((sesso= femmina) & (avatar= cartoon) & (texture= alta) & (mediaResp <= -163202)) allora numero_cluster= 5 (3.0/2.1)
- Se ((sesso= femmina) & (avatar= cartoon) & (texture= alta) & (mediaResp > -163202)) allora numero_cluster= 1 (2.0/1.8)
- Se ((sesso= femmina) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 3 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= realistico) & (mediaResp <= -48896) & (mediaGSR <= 0.014)) allora numero_cluster= 5 (4.0/2.2)

- Se ((sesso= femmina) & (avatar= realistico) & (mediaResp <= -48896) & (mediaGSR > 0.014)) allora numero_cluster= 2 (2.0/1.8)
- Se ((sesso= femmina) & (avatar= realistico) & (-48896 < mediaResp <= -11852)) allora numero_cluster= 1 (2.0/1.8)
- Se ((sesso= femmina) & (avatar= realistico) & (mediaResp>-11852)) allora numero_cluster= 0 (2.0/1.8)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaResp<=-220151)) allora numero_cluster= 2 (4.0/3.1)
- Se ((sesso= maschio) & (avatar= cartoon) & (-220151 < mediaResp <= -11852)) allora numero_cluster= 1 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaResp> -11852)) allora numero_cluster= 3 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= alta) & (mediaResp <= -208974)) allora numero_cluster= 3 (2.0/1.0)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= alta) & (mediaResp > -208974)) allora numero_cluster= 4 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= bassa)) allora numero_cluster= 4 (5.0/1.2)

C)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((mediaHR <= 76.053) & (mediaResp <= -142070)) allora numero_cluster= 0 (2.0/1.8)
- Se ((mediaHR <= 76.053) & (mediaResp > -142070)) allora numero_cluster= 3 (5.0/1.2)
- Se ((mediaHR > 76.053) & (avatar= cartoon) & (sesso= femmina) & (mediaGSR <= -0.017)) allora numero_cluster= 0 (3.0/2.1)
- Se ((mediaHR > 76.053) & (avatar= cartoon) & (sesso= femmina) & (mediaGSR > -0.017)) allora numero_cluster= 2 (6.0/2.3)
- Se ((mediaHR > 76.053) & (avatar= cartoon) & (sesso= maschio) & (mediaResp <= -250062)) allora numero_cluster= 0 (2.0/1.8)
- Se ((mediaHR > 76.053) & (avatar= cartoon) & (sesso= maschio) & (-250062< mediaResp <= -71401)) allora numero_cluster= 2 (3.0/2.1)
- Se ((mediaHR > 76.053) & (avatar= cartoon) & (sesso= maschio) & (mediaResp > -71401)) allora numero_cluster= 4 (2.0/1.0)
- Se ((mediaHR > 76.053) & (avatar= realistico) & (sesso= femmina) & (mediaResp<= -208974)) allora numero_cluster= 4 (3.0/2.1)
- Se ((mediaHR > 76.053) & (avatar= realistico) & (sesso= femmina) & (-208974 < mediaResp <= -15641)) allora numero_cluster= 1 (4.0/2.2)

- Se ((mediaHR > 76.053) & (avatar= realistico) & (sesso= femmina) & (mediaResp > -15641)) allora numero_cluster= 2 (2.0/1.8)
- Se ((mediaHR > 76.053) & (avatar= realistico) & (sesso= maschio) & (mediaGSR <= 0.038)) allora numero_cluster= 0 (5.0/1.2)
- Se ((mediaHR > 76.053) & (avatar= realistico) & (sesso= maschio) & (mediaGSR > 0.038)) allora numero_cluster= 4 (3.0/1.1)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((mediaHR <= 76.053) & (mediaResp <= -142070)) allora numero_cluster= 3 (2.0/1.8)
- Se ((mediaHR <= 76.053) & (mediaResp > -142070)) allora numero_cluster= 5 (5.0/2.3)
- Se ((mediaHR > 76.053) & (texture= alta) & (mediaGSR <= -0.043)) allora numero_cluster= 3 (3.0/2.1)
- Se ((mediaHR > 76.053) & (texture= alta) & (mediaGSR > -0.0434) & (mediaResp <= -250062)) allora numero_cluster= 0 (3.0/2.1)
- Se ((texture= alta) & (mediaGSR > -0.0434) & (-250062 < mediaResp <= -11852) & (76.053 < mediaHR <= 109.896)) allora numero_cluster=1 (7.0/2.4)
- Se ((texture= alta) & (mediaGSR > -0.0434) & (-250062 < mediaResp <= -11852) & (mediaHR > 109.896)) allora numero_cluster=0 (2.0/1.8)
- Se ((mediaHR > 76.053) & (texture= alta) & (mediaGSR > -0.0434) & (mediaResp > -11852)) allora numero_cluster= 0 (2.0/1.8)
- Se ((texture= bassa) & (sesso= femmina) & (76.053 < mediaHR <= 80.955)) allora numero_cluster= 0 (3.0/2.8)
- Se ((texture= bassa) & (sesso= femmina) & (mediaHR > 80.955)) allora numero_cluster= 4 (5.0/2.3)
- Se ((texture= bassa) & (sesso= maschio) & (76.053 < mediaHR <= 87.689)) allora numero_cluster= 0 (4.0/2.2)
- Se ((texture= bassa) & (sesso= maschio) & (mediaHR > 87.689)) allora numero_cluster= 4 (4.0/2.2)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se((sesso= femmina) & (mediaHR <= 87.689) & (mediaGSR <= 0.002)) allora numero_cluster= 4 (3.0/2.1)
- ((sesso= femmina) & (mediaHR <= 87.689) & (0.002 < mediaGSR <= 0.063)) allora numero_cluster= 2 (4.0/1.2)
- Se((sesso= femmina) & (mediaHR <= 87.689) & (mediaGSR > 0.063)) allora numero_cluster= 4 (3.0/2.1)
- Se((sesso= femmina) & (mediaHR > 87.689)) allora numero_cluster= 4 (10.0/5.6)

- Se ((sesso= maschio) & (avatar= cartoon) & (mediaHR <= 84.813) & (mediaResp <= -136168)) allora
numero_cluster= 1 (2.0/1.8)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaHR <= 84.813) & (mediaResp > -136168)) allora
numero_cluster= 2 (2.0/1.0)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaHR > 84.813) & (mediaResp <= -208974)) allora
numero_cluster= 3 (2.0/1.0)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaHR > 84.813) & (mediaResp > -208974)) allora
numero_cluster= 4 (4.0/2.2)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= alta) & (mediaResp <= -208974)) allora
numero_cluster= 2 (2.0/1.0)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= alta) & (mediaResp > -208974)) allora
numero_cluster= 3 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= bassa)) allora numero_cluster= 3
(5.0/1.2)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((sesso= femmina) & (mediaHR <= 84.813) & (avatar= cartoon) & (texture= alta)) allora
numero_cluster= 5 (2.0/1.0)
- Se ((sesso= femmina) & (avatar= cartoon) & (84.813< mediaHR <= 90.903) & (texture= alta)) allora
numero_cluster= 1 (2.0/1.0)
- Se ((sesso= femmina) & (mediaHR<= 90.903) & (avatar= cartoon) & (texture= bassa)) allora
numero_cluster= 3 (4.0/2.2)
- Se ((sesso= femmina) & (mediaHR <= 90.903) & (avatar= realistico)) allora numero_cluster= 3
(4.0/3.1)
- Se ((sesso= femmina) & (mediaHR> 90.903)) allora numero_cluster= 5 (8.0/5.4)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaHR <=84.813) & (mediaResp <= -136168)) allora
numero_cluster= 2 (2.0/1.8)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaHR<= 84.813) & (mediaResp >-136168)) allora
numero_cluster= 3 (2.0/1.0)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaHR > 84.813) & (mediaResp <= -136168)) allora
numero_cluster= 4 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= cartoon) & (mediaHR > 84.813) & (mediaResp > -136168)) allora
numero_cluster= 5 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= alta) & (mediaResp <= -208974)) allora
numero_cluster= 3 (2.0/1.0)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= alta) & (mediaResp > -208974)) allora
numero_cluster= 4 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= realistico) & (texture= bassa)) allora numero_cluster= 4
(5.0/1.2)

D)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((numero_BIPs= 0) & (mediaResp <= -109960)) allora numero_cluster= 0 (3.0/2.1)
- Se ((numero_BIPs= 0) & (mediaResp > -109960)) allora numero_cluster= 1 (2.0/1.0)
- Se (numero_BIPs= 1) allora numero_cluster= 0 (3.0/2.8)
- Se ((numero_BIPs= 2) & (mediaResp <= -11852) & (mediaGSR <= -0.02)) allora numero_cluster=0 (2.0/1.8)
- Se ((numero_BIPs= 2) & (MediaResp <= -11852) & (MediaGSR > -0.02)) allora numero_cluster= 4 (4.0/2.2)
- Se ((numero_BIPs= 2) & (mediaResp > -11852)) allora numero_cluster= 3 (2.0/1.0)
- Se ((numero_BIPs= 3) & (sesso= femmina)) allora numero_cluster= 2 (3.0/2.1)
- Se ((numero_BIPs= 3) & (sesso= maschio) & (mediaResp <= -220151)) allora numero_cluster= 0 (2.0/1.0)
- Se ((numero_BIPs= 3) & (sesso= maschio) & (mediaResp > -220151)) allora numero_cluster= 2 (2.0/1.8)
- Se ((numero_BIPs= 4) & (mediaResp <= -171942)) allora numero_cluster= 0 (2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp > -171942)) allora numero_cluster= 2 (2.0/1.0)
- Se ((numero_BIPs= 5) & (mediaResp <= -154360)) allora numero_cluster= 2 (3.0/2.1)
- Se ((numero_BIPs= 5) & (mediaResp > -154360)) allora numero_cluster= 3 (3.0/1.1)
- Se (numero_BIPs= 6) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs= 10) allora numero_cluster= 0 (1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 2 (2.0/1.8)
- Se (numero_BIPs= 16) allora numero_cluster= 3 (1.0/0.8)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((numero_BIPs= 0) & (mediaResp <= -109960)) allora numero_cluster= 3 (3.0/2.1)
- Se ((numero_BIPs= 0) & (mediaResp > -109960)) allora numero_cluster= 1 (2.0/1.0)
- Se (numero_BIPs= 1) allora numero_cluster= 0 (3.0/2.8)
- Se ((numero_BIPs= 2) & (mediaResp <= -11852) & (mediaGSR <= -0.02)) allora numero_cluster= 0 (2.0/1.8)
- Se ((numero_BIPs= 2) & (mediaResp <= -11852) & (mediaGSR > -0.02)) allora numero_cluster= 4 (4.0/2.2)
- Se ((numero_BIPs= 2) & (mediaResp > -11852)) allora numero_cluster= 5 (2.0/1.0)
- Se ((numero_BIPs= 3) & (sesso= femmina)) allora numero_cluster= 0 (3.0/2.8)
- Se ((numero_BIPs= 3) & (sesso= maschio) & (mediaResp <= -220151)) allora numero_cluster= 0 (2.0/1.0)

- Se ((numero_BIPs= 3) & (sesso= maschio) & (mediaResp > -220151)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp <= -171942)) allora numero_cluster= 0 (2.0/1.8)
- Se ((numero_BIPs= 4) & (mediaResp > -171942)) allora numero_cluster= 1 (2.0/1.8)
- Se ((numero_BIPs= 5) & (mediaResp <= -154360)) allora numero_cluster= 1 (3.0/2.1)
- Se ((numero_BIPs= 5) & (mediaResp > -154360)) allora numero_cluster= 5 (3.0/2.1)
- Se (numero_BIPs= 6) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 0 (2.0/1.8)
- Se (numero_BIPs= 10) allora numero_cluster= 0 (1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs= 16) allora numero_cluster= 2 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((numero_BIPs= 0) & (mediaGSR<= 0.038)) allora numero_cluster= 4 (3.0/2.1)
- Se ((numero_BIPs= 0) & (mediaGSR> 0.038)) allora numero_cluster= 1 (2.0/1.8)
- Se (numero_BIPs= 1) allora numero_cluster= 0 (3.0/2.8)
- Se ((numero_BIPs= 2) & (texture= alta)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp <= -11852)) allora numero_cluster= 3 (4.0/2.2)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp > -11852)) allora numero_cluster= 2 (2.0/1.0)
- Se ((numero_BIPs= 3) & (mediaResp <= -136168)) allora numero_cluster= 3 (5.0/2.3)
- Se ((numero_BIP= 3) & (mediaResp > -136168)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp <= -171942)) allora numero_cluster= 2 (2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp > -171942)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 5) & (mediaResp <= -154360)) allora numero_cluster= 4 (3.0/1.1)
- Se ((numero_BIPs= 5) & (mediaResp > -154360)) allora numero_cluster= 2 (3.0/1.1)
- Se (numero_BIPs= 6) allora numero_cluster= 3 (1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 3 (2.0/1.0)
- Se (numero_BIPs= 10) allora numero_cluster= 3 (1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((numero_BIPs= 0) & (mediaGSR <= 0.038)) allora numero_cluster= 5 (3.0/2.1)
- Se ((numero_BIPs= 0) & (mediaGSR> 0.038)) allora numero_cluster= 2 (2.0/1.8)
- Se (numero_BIPs= 1) allora numero_cluster= 0 (3.0/2.8)
- Se ((numero_BIPs= 2) & (texture= alta)) allora numero_cluster= 5 (2.0/1.0)

- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp <= -11852)) allora numero_cluster= 4
(4.0/2.2)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp > -11852)) allora numero_cluster= 3
(2.0/1.0)
- Se (numero_BIPs= 3) allora numero_cluster= 4
(7.0/4.4)
- Se ((numero_BIPs= 4) & (mediaResp <= -171942)) allora numero_cluster= 3
(2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp > -171942)) allora numero_cluster= 1
(2.0/1.8)
- Se ((numero_BIPs= 5) & (mediaResp <= -154360)) allora numero_cluster= 1
(3.0/2.1)
- Se ((numero_BIPs= 5) & (mediaResp > -154360)) allora numero_cluster= 3
(3.0/1.1)
- Se (numero_BIPs= 6) allora numero_cluster= 4
(1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 4
(2.0/1.0)
- Se (numero_BIPs= 10) allora numero_cluster= 4
(1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 5
(2.0/1.0)
- Se (numero_BIPs= 16) allora numero_cluster= 0
(1.0/0.8)

E)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((numero_BIPs= 0) & (mediaResp <= -109960)) allora numero_cluster= 0
(3.0/2.1)
- Se ((numero_BIPs= 0) & (mediaResp > -109960)) allora numero_cluster= 1
(2.0/1.0)
- Se (numero_BIPs= 1) allora numero_cluster= 0
(3.0/2.8)
- Se ((numero_BIPs= 2) & (mediaHR <= 79.204)) allora numero_cluster= 3
(3.0/2.1)
- Se ((numero_BIPs= 2) & (mediaHR > 79.204) & (mediaGSR <= -0.02)) allora numero_cluster= 0
(2.0/1.8)
- Se ((numero_BIPs= 2) & (mediaHR > 79.204) & (mediaGSR > -0.02)) allora numero_cluster= 4
(3.0/1.1)
- Se ((numero_BIPs= 3) & (mediaHR <= 83.669)) allora numero_cluster= 2
(3.0/2.1)
- Se ((numero_BIPs= 3) & (mediaHR > 83.669)) allora numero_cluster= 0
(4.0/2.2)
- Se ((numero_BIPs= 4) & (mediaResp <= -171942)) allora numero_cluster= 0
(2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp > -171942)) allora numero_cluster= 2
(2.0/1.0)
- Se ((numero_BIPs= 5) & (mediaResp <= -154360)) allora numero_cluster= 2
(3.0/2.1)
- Se ((numero_BIPs= 5) & (mediaResp > -154360)) allora numero_cluster= 3
(3.0/1.1)
- Se (numero_BIPs= 6) allora numero_cluster= 4
(1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 4
(2.0/1.0)
- Se (numero_BIPs= 10) allora numero_cluster= 0
(1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 2
(2.0/1.8)
- Se (numero_BIPs= 16) allora numero_cluster= 3
(1.0/0.8)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((numero_BIPs= 0) & (mediaResp <= -109960)) allora numero_cluster= 3 (3.0/2.1)
- Se ((numero_BIPs= 0) & (mediaResp > -109960)) allora numero_cluster= 1 (2.0/1.0)
- Se (numero_BIPs= 1) allora numero_cluster= 0 (3.0/2.8)
- Se ((numero_BIPs= 2) & (mediaHR <= 79.204)) allora numero_cluster= 5 (3.0/2.1)
- Se ((numero_BIPs= 2) & (mediaHR > 79.204) & (mediaGSR <= -0.02)) allora numero_cluster= 0 (2.0/1.8)
- Se ((numero_BIPs= 2) & (mediaHR > 79.204) & (mediaGSR > -0.02)) allora numero_cluster= 4 (3.0/1.1)
- Se ((numero_BIPs= 3) & (mediaHR<= 83.669)) allora numero_cluster= 4 (3.0/2.1)
- Se ((numero_BIPs= 3) & (mediaHR> 83.669)) allora numero_cluster= 0 (4.0/2.2)
- Se ((numero_BIPs= 4) & (mediaResp <= -171942)) allora numero_cluster= 0 (2.0/1.8)
- Se ((numero_BIPs= 4) & (mediaResp > -171942)) allora numero_cluster= 1 (2.0/1.8)
- Se ((numero_BIPs= 5) & (mediaResp <= -154360)) allora numero_cluster= 1 (3.0/2.1)
- Se ((numero_BIPs= 5) & (mediaResp > -154360)) allora numero_cluster= 5 (3.0/2.1)
- Se (numero_BIPs= 6) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 0 (2.0/1.8)
- Se (numero_BIPs= 10) allora numero_cluster= 0 (1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs= 16) allora numero_cluster= 2 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((numero_BIPs= 0) & (mediaGSR<= 0.038)) allora numero_cluster= 4 (3.0/2.1)
- Se ((numero_BIPs= 0) & (mediaGSR> 0.038)) allora numero_cluster= 1 (2.0/1.8)
- Se (numero_BIPs= 1) allora numero_cluster= 0 (3.0/2.8)
- Se ((numero_BIPs= 2) & (texture= alta)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp <= -11852)) allora numero_cluster= 3 (4.0/2.2)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp > -11852)) allora numero_cluster= 2 (2.0/1.0)
- Se ((numero_BIPs= 3) & (mediaResp <= -136168)) allora numero_cluster= 3 (5.0/2.3)
- Se ((numero_BIPs= 3) & (mediaResp > -136168)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp <= -171942)) allora numero_cluster= 2 (2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp > -171942)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 5) & (mediaResp <= -154360)) allora numero_cluster= 4 (3.0/1.1)
- Se ((numero_BIPs= 5) & (mediaResp > -154360)) allora numero_cluster= 2 (3.0/1.1)
- Se (numero_BIPs= 6) allora numero_cluster= 3 (1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 3 (2.0/1.0)

- Se (numero_BIPs= 10) allora numero_cluster= 3 (1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((numero_BIPs= 0) & (mediaGSR<= 0.038)) allora numero_cluster= 5 (3.0/2.1)
- Se ((numero_BIPs= 0) & (mediaGSR> 0.038)) allora numero_cluster= 2 (2.0/1.8)
- Se (numero_BIPs= 1) allora numero_cluster= 0 (3.0/2.8)
- Se ((numero_BIPs= 2) & (texture= alta)) allora numero_cluster= 5 (2.0/1.0)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp <= -11852)) allora numero_cluster= 4 (4.0/2.2)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp > -11852)) allora numero_cluster= 3 (2.0/1.0)
- Se ((numero_BIPs= 3) & (avatar= cartoon) & (mediaHR <= 93.983)) allora numero_cluster= 5 (2.0/1.0)
- Se ((numero_BIPs= 3) & (avatar= cartoon) & (mediaHR > 93.983)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 3) & (avatar= realistico)) allora numero_cluster= 4 (3.0/2.1)
- Se ((numero_BIPs= 4) & (mediaResp <= -171942)) allora numero_cluster= 3 (2.0/1.0)
- Se ((numero_BIPs= 4) & (mediaResp > -171942)) allora numero_cluster= 1 (2.0/1.8)
- Se ((numero_BIPs= 5) & (mediaResp <= -154360)) allora numero_cluster= 1 (3.0/2.1)
- Se ((numero_BIPs= 5) & (mediaResp > -154360)) allora numero_cluster= 3 (3.0/1.1)
- Se (numero_BIPs= 6) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs= 10) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 5 (2.0/1.0)
- Se (numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

F)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (avatar= cartoon) allora numero_cluster= 2 (20.0/13.0)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina) & (0 <= numero_BIPs<= 3)) allora numero_cluster=1 (4.0/2.2)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina) & (4 <=numero_BIPs<=7)) allora numero_cluster=0 (2.0/1.8)
- Se ((avatar=realistico) & (mediaResp <= -15641) & (sesso= femmina) & (8 <=numero_BIPs<=11)) allora numero_cluster=4 (1.0/0.8)

- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina) & (numero_BIPs= 16)) allora numero_cluster=1 (0.0)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= maschio) & (texture= alta)) allora numero_cluster=0 (3.0/1.1)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= maschio) & (texture= bassa) & (mediaGSR <= 0.038)) allora numero_cluster=0 (2.0/1.0)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= maschio) & (texture= bassa) & (mediaGSR > 0.038)) allora numero_cluster=4 (3.0/1.1)
- Se ((avatar= realistico) & (mediaResp > -15641)) allora numero_cluster= 3 (5.0/3.2)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se((texture= alta) & (0 <= numero_BIPs <= 3) & (avatar= cartoon) & (mediaGSR <= 0.038)) allora numero_cluster= 0 (4.0/3.1)
- Se((texture= alta) & (0 <= numero_BIPs <= 3) & (avatar= cartoon) & (mediaGSR > 0.038)) allora numero_cluster= 4 (2.0/1.0)
- Se((texture= alta) & (0 <= numero_BIPs <= 3) & (avatar= realistico) & (sesso= femmina)) allora numero_cluster= 1 (2.0/1.0)
- Se((texture= alta) & (0 <= numero_BIPs <= 3) & (avatar= realistico) & (sesso= maschio)) allora numero_cluster= 3 (3.0/2.1)
- Se((texture= alta) & (4 <= numero_BIPs <= 7) & (avatar= cartoon)) allora numero_cluster= 1 (4.0/2.2)
- Se((texture= alta) & (4 <= numero_BIPs <= 7) & (avatar= realistico)) allora numero_cluster= 0 (3.0/2.8)
- Se((texture= alta) & (8 <= numero_BIPs <= 11)) allora numero_cluster= 0 (1.0/0.8)
- Se((texture= alta) & (numero_BIPs= 16)) allora numero_cluster= 2 (1.0/0.8)
- Se (texture= bassa) allora numero_cluster= 4 (20.0/13.0)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -11852) & (sesso= femmina) & (avatar= cartoon)) allora numero_cluster= 4 (5.0/3.2)
- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -58495) & (sesso= femmina) & (avatar= realistico)) allora numero_cluster= 1 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 3) & (-58495 < mediaResp <= -11852) & (sesso= femmina) & (avatar= realistico)) allora numero_cluster= 4 (3.0/2.1)
- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -136168) & (sesso= maschio)) allora numero_cluster= 3 (8.0/3.5)
- Se ((0 <= numero_BIPs <= 3) & (-136168 < mediaResp <= -11852) & (sesso= maschio)) allora numero_cluster= 4 (2.0/1.0)

- Se $((0 \leq \text{numero_BIPs} \leq 3) \& (\text{mediaResp} > -11852))$ allora $\text{numero_cluster} = 2$ (3.0/2.1)
- Se $((4 \leq \text{numero_BIPs} \leq 7) \& (\text{ Sesso} = \text{femmina}) \& (\text{texture} = \text{alta}))$ allora $\text{numero_cluster} = 4$ (4.0/2.2)
- Se $((4 \leq \text{numero_BIPs} \leq 7) \& (\text{ Sesso} = \text{femmina}) \& (\text{texture} = \text{bassa}))$ allora $\text{numero_cluster} = 2$ (3.0/2.1)
- Se $((4 \leq \text{numero_BIPs} \leq 7) \& (\text{ Sesso} = \text{maschio}) \& (\text{mediaGSR} \leq -0.027))$ allora $\text{numero_cluster} = 2$ (2.0/1.0)
- Se $((4 \leq \text{numero_BIPs} \leq 7) \& (\text{ Sesso} = \text{maschio}) \& (\text{mediaGSR} > -0.027))$ allora $\text{numero_cluster} = 3$ (2.0/1.8)
- Se $((8 \leq \text{numero_BIPs} \leq 11) \& (\text{mediaGSR} \leq 0.019))$ allora $\text{numero_cluster} = 4$ (2.0/1.0)
- Se $((8 \leq \text{numero_BIPs} \leq 11) \& (\text{mediaGSR} > 0.019))$ allora $\text{numero_cluster} = 3$ (3.0/1.1)
- Se $(\text{numero_BIPs} = 16)$ allora $\text{numero_cluster} = 0$ (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e $k=6$*

- Se $((0 \leq \text{numero_BIPs} \leq 3) \& (\text{mediaResp} \leq -15641) \& (\text{ Sesso} = \text{femmina}) \& (\text{avatar} = \text{cartoon}))$ allora $\text{numero_cluster} = 5$ (5.0/3.2)
- Se $((0 \leq \text{numero_BIPs} \leq 3) \& (\text{mediaResp} \leq -58495) \& (\text{ Sesso} = \text{femmina}) \& (\text{avatar} = \text{realistico}))$ allora $\text{numero_cluster} = 2$ (2.0/1.0)
- Se $((0 \leq \text{numero_BIPs} \leq 3) \& (-58495 < \text{mediaResp} \leq -15641) \& (\text{ Sesso} = \text{femmina}) \& (\text{avatar} = \text{realistico}))$ allora $\text{numero_cluster} = 4$ (2.0/1.8)
- Se $((0 \leq \text{numero_BIPs} \leq 3) \& (\text{mediaResp} \leq -15641) \& (\text{ Sesso} = \text{maschio}))$ allora $\text{numero_cluster} = 4$ (9.0/4.5)
- Se $((0 \leq \text{numero_BIPs} \leq 3) \& (\text{mediaResp} > -15641) \& (\text{texture} = \text{alta}))$ allora $\text{numero_cluster} = 0$ (2.0/1.8)
- Se $((0 \leq \text{numero_BIPs} \leq 3) \& (\text{mediaResp} > -15641) \& (\text{texture} = \text{bassa}))$ allora $\text{numero_cluster} = 3$ (3.0/2.1)
- Se $((4 \leq \text{numero_BIPs} \leq 7) \& (\text{texture} = \text{alta}) \& (\text{avatar} = \text{cartoon}))$ allora $\text{numero_cluster} = 1$ (4.0/2.2)
- Se $((4 \leq \text{numero_BIPs} \leq 7) \& (\text{texture} = \text{alta}) \& (\text{avatar} = \text{realistico}))$ allora $\text{numero_cluster} = 3$ (3.0/2.1)
- Se $((4 \leq \text{numero_BIPs} \leq 7) \& (\text{texture} = \text{bassa}) \& (\text{mediaGSR} \leq 0.076))$ allora $\text{numero_cluster} = 3$ (2.0/1.0)
- Se $((4 \leq \text{numero_BIPs} \leq 7) \& (\text{texture} = \text{bassa}) \& (\text{mediaGSR} > 0.076))$ allora $\text{numero_cluster} = 4$ (2.0/1.8)
- Se $((8 \leq \text{numero_BIPs} \leq 11) \& (\text{mediaGSR} \leq 0.019))$ allora $\text{numero_cluster} = 5$ (2.0/1.0)
- Se $((8 \leq \text{numero_BIPs} \leq 11) \& (\text{mediaGSR} > 0.019))$ allora $\text{numero_cluster} = 4$ (3.0/1.1)
- Se $(\text{numero_BIPs} = 16)$ allora $\text{numero_cluster} = 0$ (1.0/0.8)

G)**➤ Caso: Clusters determinati con k-means e k= 5**

- Se ((avatar= cartoon) & (0 <= numero_BIPs <= 3) & (sesso= femmina) & (mediaResp<= -220151)) allora numero_cluster= 2 (2.0/1.0)
- Se ((avatar= cartoon) & (0 <= numero_BIPs <= 3) & (sesso= femmina) & (mediaResp> -220151)) allora numero_cluster= 0 (3.0/2.1)
- Se ((avatar= cartoon) & (0 <= numero_BIPs <= 3) & (sesso= maschio) & (mediaHR <= 89.948)) allora numero_cluster=1 (3.0/2.8)
- Se ((avatar= cartoon) & (0 <= numero_BIPs <= 3) & (sesso= maschio) & (mediaHR > 89.948)) allora numero_cluster=4 (3.0/2.1)
- Se ((avatar= cartoon) & (4<= numero_BIPs <= 7) & (mediaHr<= 80.955)) allora numero_cluster= 3 (3.0/1.1)
- Se ((avatar= cartoon) & (4<= numero_BIPs <= 7) & (mediaHR > 80.955)) allora numero_cluster= 2 (4.0/1.2)
- Se ((avatar= cartoon) & (8<= numero_BIPs <= 11)) allora numero_cluster= 2 (2.0/1.8)
- Se ((avatar= cartoon) & (numero_BIPs= 16)) allora numero_cluster= 2 (0.0)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina) & (0 <= numero_BIPs <= 3)) allora numero_cluster= 1 (4.0/2.2)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina) & (4<= numero_BIPs <= 7)) allora numero_cluster= 0 (2.0/1.8)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina) & (8<= numero_BIPs <= 11)) allora numero_cluster= 4 (1.0/0.8)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina) & (numero_BIPs= 16)) allora numero_cluster= 1 (0.0)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= maschio) & (texture= alta)) allora numero_cluster= 0 (3.0/1.1)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= maschio) & (texture= bassa) & (mediaGSR <= 0.038)) allora numero_cluster= 0 (2.0/1.0)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= maschio) & (texture= bassa) & (mediaGSR >0.038)) allora numero_cluster= 4 (3.0/1.1)
- Se ((avatar= realistico) & (mediaResp > -15641)) allora numero_cluster= 3 (5.0/3.2)

➤ Caso: Clusters determinati con k-means e k= 6

- Se ((mediaHR <= 76.053) & (mediaResp <= -142070)) allora numero_cluster= 3 (2.0/1.8)
- Se ((mediaHR <= 76.053) & (mediaResp > -142070)) allora numero_cluster= 5 (5.0/2.3)
- Se ((76.053< mediaHR <= 92.057) & (texture= alta) & (0 <= numero_BIPs <= 3) & (avatar= cartoon) & (mediaGSR <= 0.038)) allora numero_cluster= 1 (2.0/1.0)

- Se ((mediaHR > 92.057) & (texture= alta) & (0 <= numero_BIPs <= 3) & (avatar= cartoon) & (mediaGSR <= 0.038)) allora numero_cluster= 0 (2.0/1.0)
- Se ((mediaHR > 76.053) & (texture= alta) & (0 <= numero_BIPs <= 3) & (avatar= cartoon) & (mediaGSR > 0.038)) allora numero_cluster= 4 (2.0/1.0)
- Se ((mediaHR > 76.053) & (texture= alta) & (0 <= numero_BIPs <= 3) & (avatar= realistico)) allora numero_cluster= 1 (3.0/2.1)
- Se ((mediaHR > 76.053) & (texture= alta) & (4 <= numero_BIPs <= 7) & (avatar= cartoon)) allora numero_cluster= 1 (3.0/1.1)
- Se ((mediaHR > 76.053) & (texture= alta) & (4 <= numero_BIPs <= 7) & (avatar= realistico)) allora numero_cluster= 2 (3.0/2.8)
- Se ((mediaHR > 76.053) & (texture= alta) & (8 <= numero_BIPs <= 11)) allora numero_cluster= 0 (1.0/0.8)
- Se ((mediaHR > 76.053) & (texture= alta) & (numero_BIPs= 16)) allora numero_cluster= 2 (1.0/0.8)
- Se ((mediaHR > 76.053) & (texture= bassa)) allora numero_cluster= 4 (16.0/9.8)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -11852) & (sesso= femmina) & (avatar= cartoon)) allora numero_cluster= 4 (5.0/3.2)
- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -58495) & (sesso= femmina) & (avatar= realistico)) allora numero_cluster= 1 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 3) & (-58495 < mediaResp <= -11852) & (sesso= femmina) & (avatar= realistico)) allora numero_cluster= 4 (3.0/2.1)
- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -136168) & (sesso= maschio)) allora numero_cluster= 3 (8.0/3.5)
- Se ((0 <= numero_BIPs <= 3) & (-136168 < mediaResp <= -11852) & (sesso= maschio)) allora numero_cluster= 4 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 3) & (mediaResp > -11852)) allora numero_cluster= 2 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 7) & (mediaHR <= 87.689)) allora numero_cluster= 2 (5.0/2.3)
- Se ((4 <= numero_BIPs <= 7) & (mediaHR > 87.689)) allora numero_cluster= 4 (6.0/3.3)
- Se ((8 <= numero_BIPs <= 11) & (mediaGSR <= 0.019)) allora numero_cluster= 4 (2.0/1.0)
- Se ((8 <= numero_BIPs <= 11) & (mediaGSR > 0.019)) allora numero_cluster= 3 (3.0/1.1)
- Se (numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -15641) & (sesso= femmina) & (avatar= cartoon)) allora numero_cluster= 5 (5.0/3.2)

- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -58495) & (sesso= femmina) & (avatar= realistico)) allora numero_cluster= 2 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 3) & (-58495 < mediaResp <= -15641) & (sesso= femmina) & (avatar= realistico)) allora numero_cluster= 4 (2.0/1.8)
- Se ((0 <= numero_BIPs <= 3) & (mediaResp <= -15641) & (sesso= maschio)) allora numero_cluster= 4 (9.0/4.5)
- Se ((0 <= numero_BIPs <= 3) & (mediaResp > -15641) & (texture= alta)) allora numero_cluster= 0 (2.0/1.8)
- Se ((0 <= numero_BIPs <= 3) & (mediaResp > -15641) & (texture= bassa)) allora numero_cluster= 3 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 7) & (texture= alta) & (avatar= cartoon)) allora numero_cluster= 1 (4.0/2.2)
- Se ((4 <= numero_BIPs <= 7) & (texture= alta) & (avatar= realistico)) allora numero_cluster= 3 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 7) & (texture= bassa) & (mediaGSR <= 0.076)) allora numero_cluster= 3 (2.0/1.0)
- Se ((4 <= numero_BIPs <= 7) & (texture= 'bassa') & (mediaGSR > 0.076)) allora numero_cluster= 4 (2.0/1.8)
- Se ((8 <= numero_BIPs <= 11) & (mediaGSR <= 0.019)) allora numero_cluster= 5 (2.0/1.0)
- Se ((8 <= numero_BIPs <= 11) & (mediaGSR > 0.019)) allora numero_cluster= 4 (3.0/1.1)
- Se (numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

H)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (avatar= cartoon) allora numero_cluster= 2 (20.0/13.0)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina)) allora numero_cluster=1 (7.0/5.3)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= maschio)) allora numero_cluster=0 (8.0/4.5)
- Se ((avatar= realistico) & (mediaResp > -15641)) allora numero_cluster= 3 (5.0/3.2)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((0 <= numero_BIPs <= 1) & (texture= alta) & (sesso= femmina)) allora numero_cluster=1 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 1) & (texture= alta) & (sesso= maschio)) allora numero_cluster=3 (4.0/3.1)
- Se ((0 <= numero_BIPs <= 1) & (texture= bassa)) allora numero_cluster= 3 (2.0/1.8)

- Se ((numero_BIPs= 2) & (mediaResp <= -11852) & (mediaGSR <= -0.02)) allora numero_cluster= 0
(2.0/1.8)
- Se ((numero_BIPs= 2) & (mediaResp <= -11852) & (mediaGSR > -0.02)) allora numero_cluster= 4
(4.0/2.2)
- Se ((numero_BIPs= 2) & (mediaResp > -11852)) allora numero_cluster= 5
(2.0/1.0)
- Se ((numero_BIPs= 3) & (sesso= femmina)) allora numero_cluster= 0
(3.0/2.8)
- Se ((numero_BIPs= 3) & (sesso= maschio) & (mediaResp <= -220151)) allora numero_cluster= 0
(2.0/1.0)
- Se ((numero_BIPs= 3) & (sesso= maschio) & (mediaResp > -220151)) allora numero_cluster= 4
(2.0/1.0)
- Se ((4 <= numero_BIPs <= 5) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 1
(4.0/2.2)
- Se ((4 <= numero_BIPs <= 5) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 0
(3.0/2.8)
- Se ((4 <= numero_BIPs <= 5) & (avatar= realistico)) allora numero_cluster= 0
(3.0/2.8)
- Se ((6 <= numero_BIPs <= 16) & (mediaResp <= -109960)) allora numero_cluster= 4
(4.0/2.2)
- Se ((6 <= numero_BIPs <= 16) & (mediaResp > -109960)) allora numero_cluster= 0
(3.0/2.1)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((0 <= numero_BIPs <= 1) & (sesso= femmina)) allora numero_cluster= 4
(4.0/3.1)
- Se ((0 <= numero_BIPs <= 1) & (sesso= maschio) & (mediaResp <= -208974)) allora numero_cluster= 1
(2.0/1.8)
- Se ((0 <= numero_BIPs <= 1) & (sesso= maschio) & (mediaResp > -208974)) allora numero_cluster= 0
(2.0/1.8)
- Se ((numero_BIPs= 2) & (texture= alta)) allora numero_cluster= 4
(2.0/1.0)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp <= -11852)) allora numero_cluster= 3
(4.0/2.2)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp > -11852)) allora numero_cluster= 2
(2.0/1.0)
- Se ((numero_BIPs= 3) & (mediaResp <= -136168)) allora numero_cluster= 3
(5.0/2.3)
- Se ((numero_BIPs= 3) & (mediaResp > -136168)) allora numero_cluster= 4
(2.0/1.0)
- Se ((4 <= numero_BIPs <= 5) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 4
(4.0/2.2)
- Se ((4 <= numero_BIPs <= 5) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 2
(3.0/2.1)
- Se ((4 <= numero_BIPs <= 5) & (sesso= maschio)) allora numero_cluster= 2
(3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (mediaGSR <= 0.02)) allora numero_cluster= 4
(3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (mediaGSR > 0.02)) allora numero_cluster= 3
(4.0/1.2)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((0 <= numero_BIPs <= 1) & (sesso= femmina)) allora numero_cluster= 5 (4.0/3.1)
- Se ((0 <= numero_BIPs <= 1) & (sesso= maschio) & (mediaResp <= -208974)) allora numero_cluster= 2 (2.0/1.8)
- Se ((0 <= numero_BIPs <= 1) & (sesso= maschio) & (mediaResp > -208974)) allora numero_cluster= 0 (2.0/1.8)
- Se ((numero_BIPs= 2) & (texture= alta)) allora numero_cluster= 5 (2.0/1.0)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp <= -11852)) allora numero_cluster= 4 (4.0/2.2)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp > -11852)) allora numero_cluster= 3 (2.0/1.0)
- Se (numero_BIPs= 3) allora numero_cluster= 4 (7.0/4.4)
- Se ((4 <= numero_BIPs <= 5) & (texture= alta) & (avatar= cartoon)) allora numero_cluster= 1 (4.0/2.2)
- Se ((4 <= numero_BIPs <= 5) & (texture= alta) & (avatar= realistico)) allora numero_cluster= 3 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 5) & (texture= bassa)) allora numero_cluster= 3 (3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (mediaGSR <= 0.02)) allora numero_cluster= 5 (3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (mediaGSR > 0.02)) allora numero_cluster= 4 (4.0/1.2)

I)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((avatar= cartoon) & (0 <= numero_BIPs <= 1)) allora numero_cluster= 0 (3.0/2.8)
- Se ((avatar= cartoon) & (numero_BIPs= 2) & (mediaHR <= 83.669)) allora numero_cluster= 2 (2.0/1.8)
- Se ((avatar= cartoon) & (numero_BIPs= 2) & (mediaHR > 83.669)) allora numero_cluster= 4 (2.0/1.0)
- Se ((avatar= cartoon) & (numero_BIPs= 3) & (mediaHR <= 93.983)) allora numero_cluster= 2 (2.0/1.0)
- Se ((avatar= cartoon) & (numero_BIPs= 3) & (mediaHR > 93.983)) allora numero_cluster= 0 (2.0/1.0)
- Se ((avatar= cartoon) & (4 <= numero_BIPs <= 5) & (mediaHR <= 80.955)) allora numero_cluster= 3 (3.0/1.1)
- Se ((avatar= cartoon) & (4 <= numero_BIPs <= 5) & (mediaHR > 80.955)) allora numero_cluster= 2 (4.0/1.2)
- Se ((avatar= cartoon) & (6 <= numero_BIPs <= 16)) allora numero_cluster= 2 (2.0/1.8)
- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= femmina)) allora numero_cluster= 1 (7.0/5.3)

- Se ((avatar= realistico) & (mediaResp <= -15641) & (sesso= maschio)) allora numro_cluster= 0 (8.0/4.5)

- Se ((avatar= realistico) & (mediaResp > -15641)) allora numero_cluster= 3 (5.0/3.2)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se (mediaHR <= 76.053) allora numero_cluster= 5 (7.0/4.4)
- Se ((mediaHR > 76.053) & (0 <= numero_BIPs <= 1) & (texture= alta)) allora numero_cluster= 1 (4.0/2.2)
- Se ((mediaHR > 76.053) & (0 <= numero_BIPs <= 1) & (texture= bassa)) allora numero_cluster= 3 (2.0/1.8)
- Se((mediaHR > 76.053) & (numero_BIPs= 2) & (mediaGSR<= -0.02)) allora numero_cluster= 0 (2.0/1.8)
- Se((mediaHR > 76.053) & (numero_BIPs= 2) & (mediaGSR> -0.02)) allora numero_cluster= 4 (4.0/2.2)
- Se ((76.053 < mediaHR <=83.669) & (numero_BIPs= 3)) allora numero_cluster= 4 (3.0/2.1)
- Se ((mediaHR > 83.669) & (numero_BIPs= 3)) allora numero_cluster= 0 (4.0/2.2)
- Se ((mediaHR > 76.053) & (4 <= numero_BIPs <= 5) & (sesso= femmina) & (mediaGSR <= 0.03)) allora numero_cluster= 0 (3.0/2.1)
- Se ((mediaHR > 76.053) & (4 <= numero_BIPs <= 5) & (sesso= femmina) & (mediaGSR > 0.03)) allora numero_cluster= 1 (3.0/2.1)
- Se ((mediaHR > 76.053) & (4 <= numero_BIPs <= 5) & (sesso= maschio)) allora numero_cluster= 1 (2.0/1.8)
- Se ((mediaHR > 76.053) & (6 <= numero_BIPs <= 16) & (mediaResp <= -109960)) allora numero_cluster= 4 (3.0/2.1)
- Se ((mediaHR > 76.053) & (6 <= numero_BIPs <= 16) & (mediaResp > -109960)) allora numero_cluster= 0 (3.0/2.1)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se((0 <= numero_BIPs <= 1) & (mediaHR <= 77.078)) allora numero_cluster= 0 (2.0/1.8)
- Se ((0 <= numero_BIPs <= 1) & (mediaHR > 77.078) & (sesso= femmina)) allora numero_cluster= 4 (4.0/3.1)
- Se ((0 <= numero_BIPs <= 1) & (mediaHR > 77.078) & (sesso= maschio)) allora numero_cluster= 1 (2.0/1.8)
- Se ((numero_BIPs= 2) & (texture= alta)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp <= -11852)) allora numero_cluster= 3 (4.0/2.2)

- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp > -11852)) allora numero_cluster= 2 (2.0/1.0)
- Se ((numero_BIPs= 3) & (mediaResp <= -136168)) allora numero_cluster= 3 (5.0/2.3)
- Se ((numero_BIPs= 3) & (mediaResp > -136168)) allora numero_cluster= 4 (2.0/1.0)
- Se ((4 <= numero_BIPs <= 5) & (mediaHR <= 80.955)) allora numero_cluster= 2 (3.0/1.1)
- Se ((4 <= numero_BIPs <= 5) & (mediaHR > 80.955) & (mediaResp <= -270706)) allora numero_cluster= 2 (2.0/1.0)
- Se ((4 <= numero_BIPs <= 5) & (mediaHR > 80.955) & (mediaResp > -270706)) allora numero_cluster= 4 (5.0/1.2)
- Se ((6 <= numero_BIPs <= 16) & (mediaGSR <= 0.019)) allora numero_cluster= 4 (3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (mediaGSR > 0.019)) allora numero_cluster= 3 (4.0/1.2)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se((0 <= numero_BIPs <= 1) & (mediaHR <= 77.078)) allora numero_cluster= 0 (2.0/1.8)
- Se ((0 <= numero_BIPs <= 1) & (mediaHR > 77.078) & (sesso= femmina)) allora numero_cluster= 5 (4.0/3.1)
- Se ((0 <= numero_BIPs <= 1) & (mediaHR > 77.078) & (sesso= maschio)) allora numero_cluster= 2 (2.0/1.8)
- Se ((numero_BIPs= 2) & (texture= alta)) allora numero_cluster= 5 (2.0/1.0)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp <= -11852)) allora numero_cluster= 4 (4.0/2.2)
- Se ((numero_BIPs= 2) & (texture= bassa) & (mediaResp > -11852)) allora numero_cluster= 3 (2.0/1.0)
- Se ((numero_BIPs= 3) & (avatar= cartoon) & (mediaHR <= 93.983)) allora numero_cluster= 5 (2.0/1.0)
- Se ((numero_BIPs= 3) & (avatar= cartoon) & (mediaHR > 93.983)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 3) & (avatar= realistico)) allora numero_cluster= 4 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 5) & (texture= alta) & (avatar= cartoon)) allora numero_cluster= 1 (4.0/2.2)
- Se ((4 <= numero_BIPs <= 5) & (texture= alta) & (avatar= realistico)) allora numero_cluster= 3 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 5) & (texture= bassa)) allora numero_cluster= 3 (3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (mediaGSR <= 0.019)) allora numero_cluster= 5 (3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (mediaGSR > 0.019)) allora numero_cluster= 4 (4.0/1.2)

L)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((age= 19) & (mediaResp <= -190248)) allora numero_cluster= 0 (2.0/1.8)
- Se ((age= 19) & (MediaResp > -190248)) allora numero_cluster= 1 (2.0/1.8)
- Se (age= 20) allora numero_cluster= 1 (3.0/2.8)
- Se (age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (age= 22) allora numero_cluster= 0 (1.0/0.8)
- Se (age= 23) allora numero_cluster= 3 (2.0/1.8)
- Se (age= 24) allora numero_cluster= 0 (4.0/2.2)
- Se (age= 25) allora numero_cluster= 3 (3.0/2.1)
- Se (age= 26) allora numero_cluster= 0 (2.0/1.8)
- Se (age= 27) allora numero_cluster= 3 (2.0/1.8)
- Se (age= 28) allora numero_cluster= 2 (3.0/2.1)
- Se (age= 29) allora numero_cluster= 3 (1.0/0.8)
- Se (age= 30) allora numero_cluster= 0 (2.0/1.8)
- Se (age= 32) allora numero_cluster= 4 (1.0/0.8)
- Se (age= 34) allora numero_cluster= 2 (2.0/1.8)
- Se (age= 35) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 40) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 41) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 43) allora numero_cluster= 2 (1.0/0.8)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((Games= 1) & (VirtualReality= 1) & (mediaResp <= -208974)) allora numero_cluster= 4 (3.0/1.1)
- Se ((Games= 1) & (VirtualReality= 1) & (mediaResp > -208974)) allora numero_cluster= 1 (4.0/2.2)
- Se ((Games= 1) & (Virtual Reality= 2)) allora numero_cluster= 4 (2.0/1.8)
- Se ((Games= 1) & (Virtual Reality= 3)) allora numero_cluster= 4 (0.0)
- Se ((Games= 1) & (Virtual Reality= 4)) allora numero_cluster= 3 (1.0/0.8)
- Se ((Games= 1) & (Virtual Reality= 5)) allora numero_cluster= 4 (0.0)
- Se (Games= 2) allora numero_cluster= 4 (13.0/8.7)
- Se (Games= 3) allora numero_cluster= 4 (1.0/0.8)
- Se ((Games= 4) & (mediaResp <= -208974)) allora numero_cluster= 3 (2.0/1.0)
- Se ((Games= 4) & (mediaResp > -208974)) allora numero_cluster= 2 (2.0/1.8)
- Se (Games= 5) allora numero_cluster= 1 (2.0/1.8)
- Se (Games= 6) allora numero_cluster= 0 (2.0/1.0)

- Se ((Games= 7) & (Occupy= 1)) allora numero_cluster= 0 (2.0/1.0)
- Se ((Games= 7) & (Occupy= 2) & (mediaResp <= -71401)) allora numero_cluster= 3 (2.0/1.0)
- Se ((Games= 7) & (Occupy= 2) & (mediaResp > -71401)) allora numero_cluster= 0 (3.0/1.1)
- Se ((Games= 7) & (Occupy= 3)) allora numero_cluster= 0 (0.0)
- Se ((Games= 7) & (Occupy= 6)) allora numero_cluster= 0 (0.0)
- Se ((Games= 7) & (Occupy= 8)) allora numero_cluster= 5 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (Gametimes= 1) allora numero_cluster= 4 (21.0/11.1)
- Se ((Gametimes= 2) & (Literate= 3)) allora numero_cluster= 1 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 4)) allora numero_cluster= 4 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 5)) allora numero_cluster= 2 (3.0/1.1)
- Se ((Gametimes= 2) & (Literate= 6)) allora numero_cluster= 2 (0.0)
- Se ((Gametimes= 2) & (Literate= 7)) allora numero_cluster= 2 (0.0)
- Se (Gametimes= 3) allora numero_cluster= 3 (5.0/2.3)
- Se (Gametimes= 4) allora numero_cluster= 2 (2.0/1.8)
- Se (Gametimes= 5) allora numero_cluster= 3 (1.0/0.8)
- Se (Gametimes= 6) allora numero_cluster= 0 (1.0/0.8)
- Se (Gametimes= 7) allora numero_cluster= 3 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (Gametimes= 1) allora numero_cluster= 5 (21.0/13.0)
- Se ((Gametimes= 2) & (Literate= 3)) allora numero_cluster= 2 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 4)) allora numero_cluster= 1 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 5)) allora numero_cluster= 3 (3.0/1.1)
- Se ((Gametimes= 2) & (Literate= 6)) allora numero_cluster= 3 (0.0)
- Se ((Gametimes= 2) & (Literate= 7)) allora numero_cluster= 3 (0.0)
- Se (Gametimes= 3) allora numero_cluster= 4 (5.0/2.3)
- Se (Gametimes= 4) allora numero_cluster= 3 (2.0/1.8)
- Se (Gametimes= 5) allora numero_cluster= 4 (1.0/0.8)
- Se (Gametimes= 6) allora numero_cluster= 0 (1.0/0.8)
- Se (Gametimes= 7) allora numero_cluster= 4 (1.0/0.8)

M)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((age= 19) & (mediaResp <= -190248)) allora numero_cluster= 0 (2.0/1.8)
- Se ((age= 19) & (MediaResp > -190248)) allora numero_cluster= 1 (2.0/1.8)

- Se (age= 20) allora numero_cluster= 1 (3.0/2.8)
- Se (age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (age= 22) allora numero_cluster= 0 (1.0/0.8)
- Se (age= 23) allora numero_cluster= 3 (2.0/1.8)
- Se (age= 24) allora numero_cluster= 0 (4.0/2.2)
- Se (age= 25) allora numero_cluster= 3 (3.0/2.1)
- Se (age= 26) allora numero_cluster= 0 (2.0/1.8)
- Se (age= 27) allora numero_cluster= 3 (2.0/1.8)
- Se (age= 28) allora numero_cluster= 2 (3.0/2.1)
- Se (age= 29) allora numero_cluster= 3 (1.0/0.8)
- Se (age= 30) allora numero_cluster= 0 (2.0/1.8)
- Se (age= 32) allora numero_cluster= 4 (1.0/0.8)
- Se (age= 34) allora numero_cluster= 2 (2.0/1.8)
- Se (age= 35) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 40) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 41) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 43) allora numero_cluster= 2 (1.0/0.8)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((Games= 1) & (VirtualReality= 1) & (mediaResp <= -208974)) allora numero_cluster= 4 (3.0/1.1)
- Se ((Games= 1) & (VirtualReality= 1) & (mediaResp > -208974)) allora numero_cluster= 1 (4.0/2.2)
- Se ((Games= 1) & (Virtual Reality= 2)) allora numero_cluster= 4 (2.0/1.8)
- Se ((Games= 1) & (Virtual Reality= 3)) allora numero_cluster= 4 (0.0)
- Se ((Games= 1) & (Virtual Reality= 4)) allora numero_cluster= 3 (1.0/0.8)
- Se ((Games= 1) & (Virtual Reality= 5)) allora numero_cluster= 4 (0.0)
- Se (Games= 2) allora numero_cluster= 4 (13.0/8.7)
- Se (Games= 3) allora numero_cluster= 4 (1.0/0.8)
- Se ((Games= 4) & (mediaResp <= -208974)) allora numero_cluster= 3 (2.0/1.0)
- Se ((Games= 4) & (mediaResp > -208974)) allora numero_cluster= 2 (2.0/1.8)
- Se (Games= 5) allora numero_cluster= 1 (2.0/1.8)
- Se (Games= 6) allora numero_cluster= 0 (2.0/1.0)
- Se ((Games= 7) & (Occupy= 1)) allora numero_cluster= 0 (2.0/1.0)
- Se ((Games= 7) & (Occupy= 2) & (mediaResp <= -71401)) allora numero_cluster= 3 (2.0/1.0)
- Se ((Games= 7) & (Occupy= 2) & (mediaResp > -71401)) allora numero_cluster= 0 (3.0/1.1)
- Se ((Games= 7) & (Occupy= 3)) allora numero_cluster= 0 (0.0)
- Se ((Games= 7) & (Occupy= 6)) allora numero_cluster= 0 (0.0)
- Se ((Games= 7) & (Occupy= 8)) allora numero_cluster= 5 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (Gametimes= 1) allora numero_cluster= 4 (21.0/11.1)
- Se ((Gametimes= 2) & (Literate= 3)) allora numero_cluster= 1 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 4)) allora numero_cluster= 4 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 5)) allora numero_cluster= 2 (3.0/1.1)
- Se ((Gametimes= 2) & (Literate= 6)) allora numero_cluster= 2 (0.0)
- Se ((Gametimes= 2) & (Literate= 7)) allora numero_cluster= 2 (0.0)
- Se (Gametimes= 3) allora numero_cluster= 3 (5.0/2.3)
- Se (Gametimes= 4) allora numero_cluster= 2 (2.0/1.8)
- Se (Gametimes= 5) allora numero_cluster= 3 (1.0/0.8)
- Se (Gametimes= 6) allora numero_cluster= 0 (1.0/0.8)
- Se (Gametimes= 7) allora numero_cluster= 3 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (Gametimes= 1) allora numero_cluster= 5 (21.0/13.0)
- Se ((Gametimes= 2) & (Literate= 3)) allora numero_cluster= 2 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 4)) allora numero_cluster= 1 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 5)) allora numero_cluster= 3 (3.0/1.1)
- Se ((Gametimes= 2) & (Literate= 6)) allora numero_cluster= 3 (0.0)
- Se ((Gametimes= 2) & (Literate= 7)) allora numero_cluster= 3 (0.0)
- Se (Gametimes= 3) allora numero_cluster= 4 (5.0/2.3)
- Se (Gametimes= 4) allora numero_cluster= 3 (2.0/1.8)
- Se (Gametimes= 5) allora numero_cluster= 4 (1.0/0.8)
- Se (Gametimes= 6) allora numero_cluster= 0 (1.0/0.8)
- Se (Gametimes= 7) allora numero_cluster= 4 (1.0/0.8)

N)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((age= 19) & (mediaResp <= -190248)) allora numero_cluster= 0 (2.0/1.8)
- Se ((age= 19) & (MediaResp > -190248)) allora numero_cluster= 1 (2.0/1.8)
- Se (age= 20) allora numero_cluster= 1 (3.0/2.8)
- Se (age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (age= 22) allora numero_cluster= 0 (1.0/0.8)
- Se (age= 23) allora numero_cluster= 3 (2.0/1.8)
- Se (age= 24) allora numero_cluster= 0 (4.0/2.2)
- Se (age= 25) allora numero_cluster= 3 (3.0/2.1)

- Se (age= 26) allora numero_cluster= 0 (2.0/1.8)
- Se (age= 27) allora numero_cluster= 3 (2.0/1.8)
- Se (age= 28) allora numero_cluster= 2 (3.0/2.1)
- Se (age= 29) allora numero_cluster= 3 (1.0/0.8)
- Se (age= 30) allora numero_cluster= 0 (2.0/1.8)
- Se (age= 32) allora numero_cluster= 4 (1.0/0.8)
- Se (age= 34) allora numero_cluster= 2 (2.0/1.8)
- Se (age= 35) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 40) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 41) allora numero_cluster= 2 (1.0/0.8)
- Se (age= 43) allora numero_cluster= 2 (1.0/0.8)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((Games= 1) & (VirtualReality= 1) & (mediaResp <= -208974)) allora numero_cluster= 4 (3.0/1.1)
- Se ((Games= 1) & (VirtualReality= 1) & (mediaResp > -208974)) allora numero_cluster= 1 (4.0/2.2)
- Se ((Games= 1) & (Virtual Reality= 2)) allora numero_cluster= 4 (2.0/1.8)
- Se ((Games= 1) & (Virtual Reality= 3)) allora numero_cluster= 4 (0.0)
- Se ((Games= 1) & (Virtual Reality= 4)) allora numero_cluster= 3 (1.0/0.8)
- Se ((Games= 1) & (Virtual Reality= 5)) allora numero_cluster= 4 (0.0)
- Se (Games= 2) allora numero_cluster= 4 (13.0/8.7)
- Se (Games= 3) allora numero_cluster= 4 (1.0/0.8)
- Se ((Games= 4) & (mediaResp <= -208974)) allora numero_cluster= 3 (2.0/1.0)
- Se ((Games= 4) & (mediaResp > -208974)) allora numero_cluster= 2 (2.0/1.8)
- Se (Games= 5) allora numero_cluster= 1 (2.0/1.8)
- Se (Games= 6) allora numero_cluster= 0 (2.0/1.0)
- Se ((Games= 7) & (Occupy= 1)) allora numero_cluster= 0 (2.0/1.0)
- Se ((Games= 7) & (Occupy= 2) & (mediaResp <= -71401)) allora numero_cluster= 3 (2.0/1.0)
- Se ((Games= 7) & (Occupy= 2) & (mediaResp > -71401)) allora numero_cluster= 0 (3.0/1.1)
- Se ((Games= 7) & (Occupy= 3)) allora numero_cluster= 0 (0.0)
- Se ((Games= 7) & (Occupy= 6)) allora numero_cluster= 0 (0.0)
- Se ((Games= 7) & (Occupy= 8)) allora numero_cluster= 5 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (Gametimes= 1) allora numero_cluster= 4 (21.0/11.1)
- Se ((Gametimes= 2) & (Literate= 3)) allora numero_cluster= 1 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 4)) allora numero_cluster= 4 (3.0/2.1)

- Se ((Gametimes= 2) & (Literate= 5)) allora numero_cluster= 2 (3.0/1.1)
- Se ((Gametimes= 2) & (Literate= 6)) allora numero_cluster= 2 (0.0)
- Se ((Gametimes= 2) & (Literate= 7)) allora numero_cluster= 2 (0.0)
- Se (Gametimes= 3) allora numero_cluster= 3 (5.0/2.3)
- Se (Gametimes= 4) allora numero_cluster= 2 (2.0/1.8)
- Se (Gametimes= 5) allora numero_cluster= 3 (1.0/0.8)
- Se (Gametimes= 6) allora numero_cluster= 0 (1.0/0.8)
- Se (Gametimes= 7) allora numero_cluster= 3 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (Gametimes= 1) allora numero_cluster= 5 (21.0/13.0)
- Se ((Gametimes= 2) & (Literate= 3)) allora numero_cluster= 2 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 4)) allora numero_cluster= 1 (3.0/2.1)
- Se ((Gametimes= 2) & (Literate= 5)) allora numero_cluster= 3 (3.0/1.1)
- Se ((Gametimes= 2) & (Literate= 6)) allora numero_cluster= 3 (0.0)
- Se ((Gametimes= 2) & (Literate= 7)) allora numero_cluster= 3 (0.0)
- Se (Gametimes= 3) allora numero_cluster= 4 (5.0/2.3)
- Se (Gametimes= 4) allora numero_cluster= 3 (2.0/1.8)
- Se (Gametimes= 5) allora numero_cluster= 4 (1.0/0.8)
- Se (Gametimes= 6) allora numero_cluster= 0 (1.0/0.8)
- Se (Gametimes= 7) allora numero_cluster= 4 (1.0/0.8)

O)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (numero_cluster= 0) allora numero_BIPs= 3 (10.0/8.3)
- Se (numero_cluster= 1) allora numero_BIPs= 0 (4.0/3.1)
- Se (numero_cluster= 2) allora numero_BIPs= 3 (10.0/8.3)
- Se (numero_cluster= 3) allora numero_BIPs= 5 (7.0/5.3)
- Se (numero_cluster= 4) allora numero_BIPs= 2 (9.0/7.3)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se (numero_cluster= 0) allora numero_BIPs= 3 (9.0/7.3)
- Se (numero_cluster= 1) allora numero_BIPs= 0 (8.0/7.1)
- Se (numero_cluster= 2) allora numero_BIPs= 2 (2.0/1.8)
- Se (numero_cluster= 3) allora numero_BIPs= 0 (5.0/4.1)
- Se (numero_cluster= 4) allora numero_BIPs= 2 (12.0/10.3)
- Se (numero_cluster= 5) allora numero_BIPs= 2 (4.0/3.1)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora numero_BIPs= 1 (2.0/1.8)
- Se (numero_cluster= 1) allora numero_BIPs= 0 (3.0/2.8)
- Se (numero_cluster= 2) allora numero_BIPs= 5 (9.0/7.3)
- Se (numero_cluster= 3) allora numero_BIPs= 3 (12.0/9.5)
- Se (numero_cluster= 4) allora numero_BIPs= 3 (14.0/12.3)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora numero_BIPs= 1 (2.0/1.8)
- Se (numero_cluster= 1) allora numero_BIPs= 5 (4.0/3.1)
- Se (numero_cluster= 2) allora numero_BIPs= 0 (3.0/2.8)
- Se (numero_cluster= 3) allora numero_BIPs= 5 (9.0/7.3)
- Se (numero_cluster= 4) allora numero_BIPs= 3 (12.0/9.5)
- Se (numero_cluster= 5) allora numero_BIPs= 0 (10.0/9.1)

P)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((numero_cluster= 0) & (mediaGSR <= -0.06)) allora numero_BIPs= 1 (2.0/1.8)
- Se ((numero_cluster= 0) & (mediaGSR > -0.06) & (mediaResp <= -269778)) allora numero_BIPs= 4 (2.0/1.0)
- Se ((numero_cluster= 0) & (-0.06 < mediaGSR <= 0.034) & (mediaResp > -269778)) allora numero_BIPs= 3 (4.0/2.2)
- Se ((numero_cluster= 0) & (mediaGSR > 0.034) & (mediaResp > -269778)) allora numero_BIPs= 0 (2.0/1.8)
- Se ((numero_cluster= 1) & (mediaResp <= -136168)) allora numero_BIPs= 1 (2.0/1.8)
- Se ((numero_cluster= 1) & (mediaResp > -136168)) allora numero_BIPs= 0 (2.0/1.0)
- Se ((numero_cluster= 2) & (texture= alta)) allora numero_BIPs= 5 (5.0/4.1)
- Se ((numero_cluster= 2) & (texture= bassa) & (mediaResp <= -208974)) allora numero_BIPs= 0 (2.0/1.8)
- Se ((numero_cluster= 2) & (texture= bassa) & (mediaResp > -208974)) allora numero_BIPs= 3 (3.0/2.1)
- Se ((numero_cluster= 3) & (avatar= cartoon)) allora numero_BIPs= 5 (4.0/2.2)
- Se ((numero_cluster= 3) & (avatar= realistico)) allora numero_BIPs= 1 (3.0/2.8)
- Se ((numero_cluster= 4) allora numero_BIPs= 2 (9.0/7.3)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((numero_cluster= 0) allora numero_BIPs= 3 (9.0/7.3)

- Se ((numero_cluster= 1) & (avatar= cartoon) & (mediaGSR <= -0.008)) allora numero_BIPs= 1
(2.0/1.8)
- Se ((numero_cluster= 1) & (avatar= cartoon) & (mediaGSR > -0.008)) allora numero_BIPs= 5
(3.0/2.1)
- Se ((numero_cluster= 1) & (avatar= realistico)) allora numero_BIPs= 0
(3.0/2.1)
- Se (numero_cluster= 2) allora numero_BIPs= 2
(2.0/1.8)
- Se ((numero_cluster= 3) & (mediaResp <= -269778)) allora numero_BIPs= 4
(2.0/1.8)
- Se ((numero_cluster= 3) & (mediaResp > -269778)) allora numero_BIPs= 0
(3.0/2.1)
- Se ((numero_cluster= 4) & (mediaResp <= -224186) & (mediaGSR <= -0.012)) allora numero_BIPs= 0
(3.0/2.8)
- Se ((numero_cluster= 4) & (mediaResp <= -224186) & (mediaGSR > -0.012)) allora numero_BIPs= 11
(3.0/2.1)
- Se ((numero_cluster= 4) & (-224186 < mediaResp <= -71401)) allora numero_BIPs= 3
(3.0/1.1)
- Se ((numero_cluster= 4) & (mediaResp > -71401)) allora numero_BIPs= 2
(3.0/2.1)
- Se (numero_cluster= 5) allora numero_BIPs= 2
(4.0/3.1)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora numero_BIPs= 1
(2.0/1.8)
- Se (numero_cluster= 1) allora numero_BIPs= 0
(3.0/2.8)
- Se ((numero_cluster= 2) & (avatar= cartoon)) allora numero_BIPs= 5
(5.0/3.2)
- Se ((numero_cluster= 2) & (avatar= realistico) & (mediaResp <= -264899)) allora numero_BIPs= 4
(2.0/1.0)
- Se ((numero_cluster= 2) & (avatar= realistico) & (mediaResp > -264899)) allora numero_BIPs= 1
(2.0/1.8)
- Se ((numero_cluster= 3) & (texture= alta) & (mediaGSR <= 0.034)) allora numero_BIPs= 3
(2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (mediaGSR > 0.034)) allora numero_BIPs= 0
(2.0/1.8)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaResp <= -250062)) allora numero_BIPs= 6
(2.0/1.8)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaResp > -250062) & (avatar= cartoon)) allora numero_BIPs= 2
(2.0/1.8)
- Se ((numero_cluster= 3) & (texture= bassa) & (-250062 < mediaResp <= -185351) & (avatar= realistico)) allora numero_BIPs= 3
(2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaResp > -185351) & (avatar= realistico)) allora numero_BIPs= 2
(2.0/1.0)

- Se ((numero_cluster= 4) & (texture= alta) & (mediaResp <= -208974)) allora numero_BIPs= 3
(3.0/2.1)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaResp > -208974)) allora numero_BIPs= 2
(5.0/4.1)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp <= -171942)) allora numero_BIPs= 11
(3.0/2.1)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp > -171942)) allora numero_BIPs= 3
(3.0/2.1)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora numero_BIPs= 1
(2.0/1.8)
- Se ((numero_cluster= 1) & (mediaResp <= -136168)) allora numero_BIPs= 5
(2.0/1.0)
- Se ((numero_cluster= 1) & (mediaResp > -136168)) allora numero_BIPs= 3
(2.0/1.8)
- Se (numero_cluster= 2) allora numero_BIPs= 0
(3.0/2.8)
- Se ((numero_cluster= 3) & (avatar= cartoon)) allora numero_BIPs= 5
(5.0/3.2)
- Se ((numero_cluster= 3) & (avatar= realistico) & (mediaResp <= -264899)) allora numero_BIPs= 4
(2.0/1.0)
- Se ((numero_cluster= 3) & (avatar= realistico) & (mediaResp > -264899)) allora numero_BIPs= 1
(2.0/1.8)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaGSR <= 0.034)) allora numero_BIPs= 3
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaGSR > 0.034)) allora numero_BIPs= 0
(2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp <= -250062)) allora numero_BIPs= 6
(2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp > -250062) & (avatar= cartoon)) allora numero_BIPs= 2
(2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (-250062 < mediaResp <= -185351) & (avatar= realistico)) allora numero_BIPs= 3
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp > -185351) & (avatar= realistico)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 2(3.0/2.1)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0
(2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (mediaResp <= -171942)) allora numero_BIPs= 11
(3.0/2.1)
- Se ((numero_cluster= 5) & (texture= bassa) & (mediaResp > -171942)) allora numero_BIPs= 3
(2.0/1.8)

Q)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((numero_cluster= 0) & (mediaHR <= 84.813)) allora numero_BIPs= 0 (3.0/2.1)
- Se ((numero_cluster=0) & (mediaHR > 84.813) & (mediaResp<= -269778)) allora numero_BIPs= 4 (2.0/1.0)
- Se ((numero_cluster=0) & (mediaHR > 84.813) & (mediaResp > -269778)) allora numero_BIPs= 3 (5.0/2.3)
- Se ((numero_cluster= 1) & (mediaResp <= -136168)) allora numero_BIPs= 1 (2.0/1.8)
- Se ((numero_cluster= 1) & (mediaResp > -136168)) allora numero_BIPs= 0 (2.0/1.0)
- Se ((numero_cluster= 2) & (texture= alta) & (mediaHR <= 84.813)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 2) & (texture= alta) & (mediaHR >84.813)) allora numero_BIPs= 5 (3.0/2.1)
- Se ((numero_cluster= 2) & (texture= bassa) & (mediaResp<= -208974)) allora numero_BIPs= 0 (2.0/1.8)
- Se ((numero_cluster= 2) & (texture= bassa) & (mediaResp > -208974)) allora numero_BIPs= 3 (3.0/2.1)
- Se ((numero_cluster= 3) & (avatar= cartoon)) allora numero_BIPs= 5 (4.0/2.2)
- Se ((numero_cluster= 3) & (avatar= realistico)) allora numero_BIPs= 1 (3.0/2.8)
- Se (numero_cluster= 4) allora numero_BIPs= 2 (9.0/7.3)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se (numero_cluster= 0) allora numero_BIPs= 3 (9.0/7.3)
- Se ((numero_cluster= 1) & (avatar= cartoon) & (mediaGSR <= -0.008)) allora numero_BIPs= 1 (2.0/1.8)
- Se ((numero_cluster= 1) & (avatar= cartoon) & (mediaGSR > -0.008)) allora numero_BIPs=5 (3.0/2.1)
- Se ((numero_cluster= 1) & (avatar= realistico)) allora numero_BIPs= 0 (3.0/2.1)
- Se (numero_cluster= 2) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 3) & (mediaResp <= -269778)) allora numero_BIPs= 4 (2.0/1.8)
- Se ((numero_cluster= 3) & (mediaResp >= -269778)) allora numero_BIPs= 0 (3.0/2.1)
- Se ((numero_cluster= 4) & (mediaResp<= -224186) & (mediaGSR <= -0.012)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 4) & (mediaResp<= -224186) & (mediaGSR > -0.012)) allora numero_BIPs= 11 (3.0/2.1)
- Se ((numero_cluster= 4) & (-224186 < mediaResp <=-71401)) allora numero_BIPs= 3 (3.0/1.1)
- Se ((numero_cluster= 4) & (mediaResp > -71401)) allora numero_BIPs= 2 (3.0/2.1)
- Se (numero_cluster= 5) allora numero_BIPs= 2 (4.0/3.1)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora numero_BIPs= 1 (2.0/1.8)
- Se (numero_cluster= 1) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 2) & (avatar= cartoon)) allora numero_BIPs= 5 (5.0/3.2)
- Se ((numero_cluster= 2) & (avatar= realistico) & (mediaResp<= -264899)) allora numero_BIPs= 4 (2.0/1.0)
- Se ((numero_cluster= 2) & (avatar= realistico) & (mediaResp> -264899)) allora numero_BIPs=1 (2.0/1.8)
- Se ((numero_cluster= 3) & (texture= alta) & (mediaGSR <= 0.034)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (mediaGSR > 0.034)) allora numero_BIPs= 0 (2.0/1.8)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaHR <= 86.867) & (mediaResp<= -185351)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaHR <= 86.867) & (mediaResp>-185351)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaHR > 86.867)) allora numero_BIPs= 8 (4.0/3.1)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaResp <= -208974)) allora numero_BIPs= 3 (3.0/2.1)
- Se((numero_cluster= 4) & (texture= alta) & (mediaResp > -208974)) allora numero_BIPs=2 (5.0/4.1)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp <= -171942)) allora numero_BIPs= 11 (3.0/2.1)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp > -171942)) allora numero_BIPs= 3 (3.0/2.1)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora numero_BIPs= 1 (2.0/1.8)
- Se ((numero_cluster=1) & (mediaResp <= -136168)) allora numero_BIPs= 5 (2.0/1.0)
- Se ((numero_cluster=1) & (mediaResp> -136168)) allora numero_BIPs=3 (2.0/1.8)
- Se (numero_cluster= 2) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 3) & (avatar= cartoon)) allora numero_BIPs= 5 (5.0/3.2)
- Se ((numero_cluster= 3) & (avatar= realistico) & (mediaResp <= -264899)) allora numero_BIPs= 4 (2.0/1.0)
- Se ((numero_cluster= 3) & (avatar= realistico) & (mediaResp > -264899)) allora numero_BIPs= 1 (2.0/1.8)

- Se ((numero_cluster= 4) & (texture= alta) & (mediaGSR <= 0.034)) allora numero_BIPs= 3
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaGSR > 0.034)) allora numero_BIPs= 0
(2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaHR <= 86.867) & (mediaResp<= -185351)) allora numero_BIPs= 3
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaHR <= 86.867) & (mediaResp> -185351)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaHR > 86.867)) allora numero_BIPs= 8
(4.0/3.1)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 2 (3.0/2.1)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0
(2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (mediaResp<= -171942)) allora numero_BIPs= 11
(3.0/2.1)
- Se ((numero_cluster= 5) & (texture= bassa) & (mediaResp > -171942)) allora numero_BIPs= 3
(2.0/1.8)

R)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((numero_cluster= 0) & (mediaResp <= -269778)) allora (4 <= numero_BIPs <= 7) (2.0/1.0)
- Se ((numero_cluster= 0) & (mediaResp > -269778)) allora (0 <= numero_BIPs <= 3) (8.0/2.4)
- Se (numero_cluster= 1) allora (0 <= numero_BIPs <= 3) (4.0/1.2)
- Se ((numero_cluster= 2) & (mediaGSR <= -0.008)) allora (0 <= numero_BIPs <= 3) (5.0/2.3)
- Se ((numero_cluster= 2) & (mediaGSR > -0.008)) allora (4 <= numero_BIPs <= 7) (5.0/2.3)
- Se ((numero_cluster= 3) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 7) (4.0/2.2)
- Se ((numero_cluster= 3) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3) (3.0/2.1)
- Se ((numero_cluster= 4) & (mediaResp <= -250062)) allora (4 <= numero_BIPs <= 7) (3.0/2.1)
- Se ((numero_cluster= 4) & (mediaResp > -250062)) allora (0 <= numero_BIPs <= 3) (6.0/3.3)

➤ *Caso: Clusters determinati con k-means e k= 6*

- (0 <= numero_BIPs <= 3) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- (0 <= numero_BIPs <= 3) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- (0 <= numero_BIPs <= 3) (40.0/19.7)

S)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((numero_cluster= 0) & (mediaResp <= -269778)) allora (4 <= numero_BIPs <= 7) (2.0/1.0)
- Se ((numero_cluster= 0) & (mediaResp > -269778)) allora (0 <= numero_BIPs <= 3) (8.0/2.4)
- Se (numero_cluster= 1) allora (0 <= numero_BIPs <= 3) (4.0/1.2)
- Se ((numero_cluster= 2) & (mediaGSR <= -0.008)) allora (0 <= numero_BIPs <= 3) (5.0/2.3)
- Se ((numero_cluster= 2) & (mediaGSR > -0.008)) allora (4 <= numero_BIPs <= 7) (5.0/2.3)
- Se ((numero_cluster= 3) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 7) (4.0/2.2)
- Se ((numero_cluster= 3) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3) (3.0/2.1)
- Se ((numero_cluster= 4) & (mediaResp <= -250062)) allora (4 <= numero_BIPs <= 7) (3.0/2.1)
- Se ((numero_cluster= 4) & (mediaResp > -250062)) allora (0 <= numero_BIPs <= 3) (6.0/3.3)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((mediaHR <= 84.81) & (texture= alta)) allora (0 <= numero_BIPs <= 3) (6.0/2.3)
- Se ((84.81 < mediaHR <= 93.13) & (texture= alta)) allora (4 <= numero_BIPs <= 7) (6.0/1.2)
- Se ((mediaHR <= 93.13) & (texture= bassa) & (sesso= femmina) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 7) (4.0/2.2)
- Se ((mediaHR <= 93.13) & (texture= bassa) & (sesso= femmina) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3) (3.0/1.1)
- Se ((mediaHR <= 93.13) & (texture= bassa) & (sesso= maschio) & (mediaResp <= -250062)) allora (4 <= numero_BIPs <= 7) (2.0/1.8)
- Se ((mediaHR <= 93.13) & (texture= bassa) & (sesso= maschio) & (mediaResp > -250062)) allora (0 <= numero_BIPs <= 3) (7.0/2.4)
- Se (mediaHR > 93.13) allora (0 <= numero_BIPs <= 3) (12.0/5.7)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- (0 <= numero_BIPs <= 3) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- (0 <= numero_BIPs <= 3) (40.0/19.7)

T)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((numero_cluster= 0) & (mediaResp <= -269778)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 0) & (mediaResp > -269778) & (mediaGSR <= -0.043)) allora (0 <= numero_BIPs <=1) (3.0/2.1)
- Se ((numero_cluster= 0) & (mediaResp > -269778) & (-0.043 < mediaGSR <= 0.034)) allora numero_BIPs= 3 (3.0/1.1)
- Se ((numero_cluster= 0) & (mediaResp > -269778) & (mediaGSR > 0.034)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se (numero_cluster= 1) allora (0 <= numero_Bips <=1) (4.0/2.2)
- Se ((numero_cluster= 2) & (mediaResp <= -264899)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se ((numero_cluster= 2) & (mediaResp > -264899) & (mediaGSR <= -0.008)) allora numero_BIPs= 3 (3.0/2.1)
- Se ((numero_cluster= 2) & (mediaResp > -264899) & (mediaGSR > -0.008)) allora (4 <= numero_BIPs <= 5) (5.0/2.3)
- Se ((numero_cluster= 3) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 5)) (4.0/2.2)
- Se ((numero_cluster= 3) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (3.0/2.8)
- Se ((numero_cluster= 4) & (texture= alta)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa)) allora (6 <= numero_BIPs <= 16) (7.0/4.4)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((numero_cluster= 0) & (mediaResp <= -71401) & (sesso= femmina)) allora (4 <= numero_BIPs <= 5) (3.0/2.1)
- Se ((numero_cluster= 0) & (mediaResp <= -71401) & (sesso= maschio)) allora numero_BIPs= 3 (3.0/2.1)
- Se ((numero_cluster= 0) & (mediaResp > -71401)) allora (6 <= numero_BIPs <= 16) (3.0/2.1)
- Se ((numero_cluster= 1) & (avatar= cartoon) & (mediaGSR <= -0.008)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se ((numero_cluster= 1) & (avatar= cartoon) & (mediaGSR > -0.008)) allora (4 <= numero_BIPs <= 5) (3.0/1.1)
- Se ((numero_cluster= 1) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se (numero_cluster= 2) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 3) & (mediaResp <= -269778)) allora (4 <= numero_BIPs <= 5) (2.0/1.8)
- Se ((numero_cluster= 3) & (mediaResp > -269778)) allora (0 <= numero_BIPs <= 1) (3.0/1.1)
- Se ((numero_cluster= 4) & (mediaResp <= -224186) & (mediaGSR <= -0.012)) allora (0 <= numero_BIPs <= 1) (3.0/2.8)

- Se ((numero_cluster= 4) & (mediaResp <= -224186) & (mediaGSR > -0.012)) allora (6 <= numero_BIPs <= 16) (3.0/1.1)
- Se ((numero_cluster= 4) & (-224186 < mediaResp <= -71401)) allora numero_BIPs= 3 (3.0/1.1)
- Se ((numero_cluster= 4) & (mediaResp > -71401)) allora numero_BIPs= 2 (3.0/2.1)
- Se (numero_cluster= 5) allora numero_BIPs= 2 (4.0/3.1)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se (numero_cluster= 1) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se ((numero_cluster= 2) & (mediaGSR <= -0.06)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 2) & (mediaGSR > -0.06) & (texture= alta)) allora (4 <= numero_BIPs <= 5) (3.0/1.1)
- Se ((numero_cluster= 2) & (mediaGSR > -0.06) & (texture= bassa) & (mediaResp <= -58495)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 2) & (mediaGSR > -0.06) & (texture= bassa) & (mediaResp > -58495)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (mediaGSR <= 0.034)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (mediaGSR > 0.034)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaResp <= -250062)) allora (6 <= numero_BIPs <= 16) (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaResp > -250062) & (avatar= cartoon)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 3) & (texture= bassa) & (-250062 < mediaResp <= -185351) & (avatar= realistico)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaResp > -185351) & (avatar= realistico)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta)) allora (4 <= numero_BIPs <= 5) (8.0/5.4)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp <= -171942)) allora (6 <= numero_BIPs <= 16) (3.0/2.1)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp > -171942)) allora numero_BIPs= 3 (3.0/2.1)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se (numero_cluster= 1) allora (4 <= numero_BIPs <= 5) (4.0/2.2)
- Se (numero_cluster= 2) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se ((numero_cluster= 3) & (mediaGSR <= -0.06)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 3) & (mediaGSR > -0.06) & (texture= alta)) allora (4 <= numero_BIPs <= 5) (3.0/1.1)
- Se ((numero_cluster= 3) & (mediaGSR > -0.06) & (texture= bassa) & (mediaResp <= -58495)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 3) & (mediaGSR > -0.06) & (texture= bassa) & (mediaResp > -58495)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaGSR <= 0.034)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaGSR > 0.034)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp <= -250062)) allora (6 <= numero_BIPs <= 16) (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp > -250062) & (avatar= cartoon)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (-250062 < mediaResp <= -185351) & (avatar= realistico)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaResp > -185351) & (avatar= realistico)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= cartoon)) allora (numero_BIPs= 2) (3.0/2.1)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (mediaResp <= -171942)) allora (6 <= numero_BIPs <= 16) (3.0/2.1)
- Se ((numero_cluster= 5) & (texture= bassa) & (mediaResp > -171942)) allora numero_BIPs= 3 (2.0/1.8)

U)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se ((numero_cluster= 0) & (mediaResp <= -269778)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 0) & (mediaResp > -269778) & (mediaGSR <= -0.043)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)

- Se ((numero_cluster= 0) & (mediaResp > -269778) & (-0.043 < mediaGSR <= 0.034)) allora numero_BIPs= 3 (3.0/1.1)
- Se ((numero_cluster= 0) & (mediaResp > -269778) & (mediaGSR > 0.034)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se (numero_cluster= 1) allora (0 <= numero_Bips <=1) (4.0/2.2)
- Se ((numero_cluster= 2) & (mediaResp <= -264899)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se ((numero_cluster= 2) & (mediaResp > -264899) & (mediaGSR <= -0.008)) allora numero_BIPs= 3 (3.0/2.1)
- Se ((numero_cluster= 2) & (mediaResp > -264899) & (mediaGSR > -0.008)) allora (4 <= numero_BIPs <= 5) (5.0/2.3)
- Se ((numero_cluster= 3) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 5)) (4.0/2.2)
- Se ((numero_cluster= 3) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (3.0/2.8)
- Se ((numero_cluster= 4) & (texture= alta)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaHR <= 84.81)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaHR > 84.81)) allora (6 <= numero_BIPs <= 16) (5.0/2.3)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se ((numero_cluster= 0) & (mediaResp <= -71401) & (sesso= femmina)) allora (4 <= numero_BIPs <= 5) (3.0/2.1)
- Se ((numero_cluster= 0) & (mediaResp <= -71401) & (sesso= maschio)) allora numero_BIPs= 3 (3.0/2.1)
- Se ((numero_cluster= 0) & (mediaResp > -71401)) allora (6 <= numero_BIPs <= 16) (3.0/2.1)
- Se ((numero_cluster= 1) & (mediaHR <= 80.95)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 1) & (mediaHR > 80.95) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 5) (4.0/2.2)
- Se ((numero_cluster= 1) & (mediaHR > 80.95) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se (numero_cluster= 2) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 3) & (mediaResp <= -269778)) allora (4 <= numero_BIPs <= 5) (2.0/1.8)
- Se ((numero_cluster= 3) & (mediaResp > -269778)) allora (0 <= numero_BIPs <= 1) (3.0/1.1)
- Se ((numero_cluster= 4) & (mediaResp <= -224186) & (mediaGSR <= -0.012)) allora (0 <= numero_BIPs <= 1) (3.0/2.8)
- Se ((numero_cluster= 4) & (mediaResp <= -224186) & (mediaGSR > -0.012)) allora (6 <= numero_BIPs <= 16) (3.0/1.1)

- Se ((numero_cluster= 4) & (-224186 < mediaResp <= -71401)) allora numero_BIPs= 3 (3.0/1.1)
- Se ((numero_cluster= 4) & (mediaResp > -71401)) allora numero_BIPs= 2 (3.0/2.1)
- Se (numero_cluster= 5) allora numero_BIPs= 2 (4.0/3.1)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se (numero_cluster= 1) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se ((numero_cluster= 2) & (mediaGSR <= -0.06)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 2) & (mediaGSR > -0.06) & (texture= alta)) allora (4 <= numero_BIPs <= 5) (3.0/1.1)
- Se ((numero_cluster= 2) & (mediaGSR > -0.06) & (texture= bassa) & (mediaResp <= -58495)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 2) & (mediaGSR > -0.06) & (texture= bassa) & (mediaResp > -58495)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (mediaGSR <= 0.034)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (mediaGSR > 0.034)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaHR <= 86.87) & (mediaResp <= -185351)) allora (numero_BIPs= 3) (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaHR <= 86.87) & (mediaResp > -185351)) allora (numero_BIPs= 2) (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (mediaHR > 86.87)) allora (6<= numero_BIPs <= 16) (4.0/2.2)
- Se ((numero_cluster= 4) & (mediaHR <= 84.81) & (texture= alta)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 4) & (84.81 < media_HR <= 95.19) & (texture= alta)) allora (4 <= numero_BIPs <= 5) (4.0/1.2)
- Se ((numero_cluster= 4) & (mediaHR <= 95.19) & (texture= bassa) & (mediaResp <= -171942)) allora (6 <= numero_BIPs <= 16) (2.0/1.0)
- Se ((numero_cluster= 4) & (mediaHR <= 95.19) & (texture= bassa) & (mediaResp > -171942)) allora numero_BIPs= 3 (3.0/2.1)
- Se ((numero_cluster= 4) & (mediaHR > 95.19)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se (numero_cluster= 1) allora (4 <= numero_BIPs <= 5) (4.0/2.2)

- Se (numero_cluster= 2) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se ((numero_cluster= 3) & (mediaGSR <= -0.06)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 3) & (mediaGSR > -0.06) & (texture= alta)) allora (4 <= numero_BIPs <= 5) (3.0/1.1)
- Se ((numero_cluster= 3) & (mediaGSR > -0.06) & (texture= bassa) & (mediaResp <= -58495)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 3) & (mediaGSR > -0.06) & (texture= bassa) & (mediaResp > -58495)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaGSR <= 0.034)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (mediaGSR > 0.034)) allora (0 <= numero_BIPs <= 1) (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaHR <= 86.87) & (mediaResp <= -185351)) allora (numero_BIPs= 3) (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaHR <= 86.87) & (mediaResp > -185351)) allora (numero_BIPs= 2) (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (mediaHR > 86.87)) allora (6<= numero_BIPs <= 16) (4.0/2.2)
- Se ((numero_cluster= 5) & (mediaHR <= 95.19) & (texture= alta)) allora numero_BIPs= 2 (3.0/2.8)
- Se ((numero_cluster= 5) & (mediaHR <= 95.19) & (texture= bassa) & (mediaResp <= -171942)) allora (6 <= numero_BIPs <= 16) (2.0/1.0)
- Se ((numero_cluster= 5) & (mediaHR <= 95.19) & (texture= bassa) & (mediaResp > -171942)) allora numero_BIPs= 3 (2.0/1.8)
- Se ((numero_cluster= 5) & (mediaHR > 95.19)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)

V)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (mediaResp <= -269778) allora numero_BIPs= 0 (7.0/6.1)
- Se (mediaResp > -269778) allora numero_BIPs= 1 (33.0/27.0)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se (mediaResp <= -269778) allora numero_BIPs= 0 (7.0/6.1)
- Se (mediaResp > -269778) allora numero_BIPs= 1 (33.0/27.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (mediaResp <= -269778) allora numero_BIPs= 0 (7.0/6.1)

- Se (mediaResp > -269778) allora numero_BIPs= 1 (33.0/27.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (mediaResp <= -269778) allora numero_BIPs= 0 (7.0/6.1)
- Se (mediaResp > -269778) allora numero_BIPs= 1 (33.0/27.0)

Z)

➤ *Caso: Clusters determinati con k-means e k= 5*

- $0 \leq \text{numero_BIPs} \leq 3$ (40.0/19.7)

➤ *Caso: Clusters determinati con k-means e k= 6*

- $0 \leq \text{numero_BIPs} \leq 3$ (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- $0 \leq \text{numero_BIPs} \leq 3$ (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- $0 \leq \text{numero_BIPs} \leq 3$ (40.0/19.7)

AA)

➤ *Caso: Clusters determinati con k-means e k= 5*

- Se (mediaGSR <= -0.61) allora ($0 \leq \text{numero_BIPs} \leq 1$) (6.0/3.3)
- Se (mediaGSR > -0.61) allora (numero_BIPs= 3) (34.0/26.2)

➤ *Caso: Clusters determinati con k-means e k= 6*

- Se (mediaGSR <= -0.61) allora ($0 \leq \text{numero_BIPs} \leq 1$) (6.0/3.3)
- Se (mediaGSR > -0.61) allora (numero_BIPs= 3) (34.0/26.2)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (mediaGSR <= -0.61) allora ($0 \leq \text{numero_BIPs} \leq 1$) (6.0/3.3)
- Se (mediaGSR > -0.61) allora (numero_BIPs= 3) (34.0/26.2)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (mediaGSR <= -0.61) allora ($0 \leq \text{numero_BIPs} \leq 1$) (6.0/3.3)
- Se (mediaGSR > -0.61) allora (numero_BIPs= 3) (34.0/26.2)

Approccio ClusterNotCluster

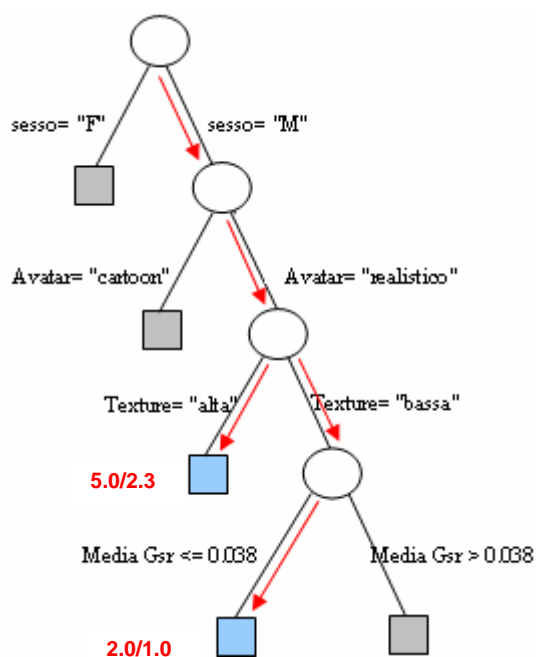
Nel paragrafo 6.7.4 è descritto un approccio che è stato utilizzato per ottenere regole di decisione caratterizzate da una confidenza più alta rispetto a quella delle regole estratte con l'approccio generale. Di seguito sono riportati gli alberi di decisione estratti con il processo di classificazione e le regole estratte da questi.

Le classificazioni sono state realizzate considerando sia i clusters determinati con il k-means sia quelli determinati con l'algoritmo gerarchico.

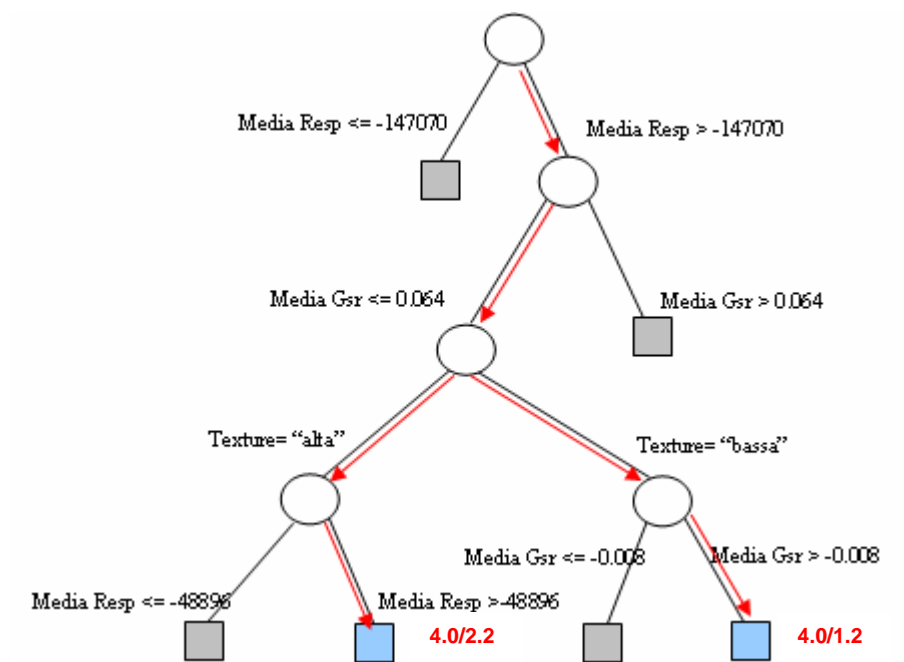
AB)

➤ *Caso: Clusters determinati con k-means e k= 5*

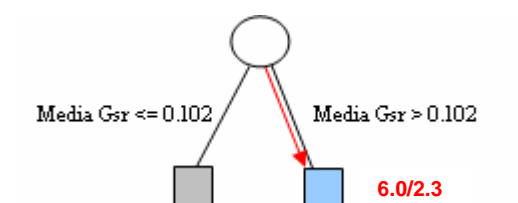
- **Cluster 0**



- Cluster 3

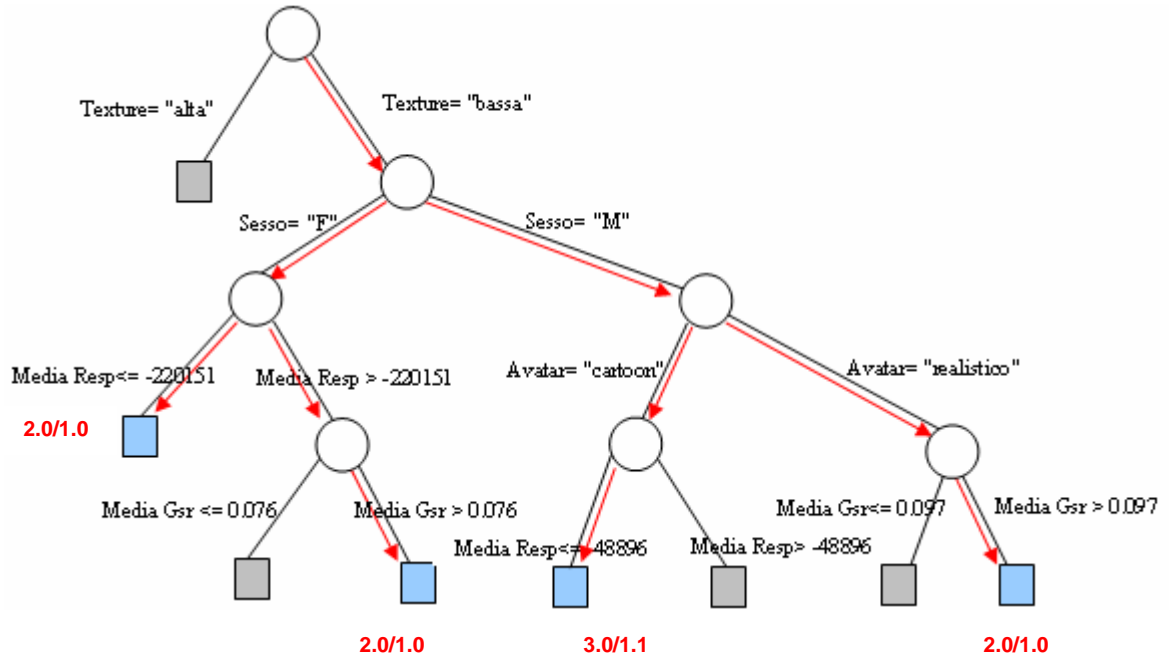


- Cluster 4



➤ *Caso: Clusters determinati con k-means e k= 6*

• **Cluster 4**



➤ *Caso: Clusters determinati con gerarchico e k= 5*

• **Cluster 3**

- Se ((sesso= maschio) & (texture= alta) & (mediaGSR > 0.019)) allora numero_cluster= 3 (4.0/2.2)
- Se ((sesso= maschio) & (texture= bassa) & (avatar= realistico)) allora numero_cluster= 3 (5.0/1.2)

• **Cluster 4**

- Se ((sesso= femmina) & (texture= alta) & (avatar= cartoon)) allora numero_cluster= 4 (5.0/2.3)
- Se ((sesso= femmina) & (texture= alta) & (avatar= realistico) & (mediaGSR ≤ 0.014)) allora numero_cluster= 4 (3.0/2.1)
- Se ((sesso= femmina) & (texture= bassa) & (mediaResp ≤ -220151)) allora numero_cluster= 4 (2.0/1.0)
- Se ((sesso= femmina) & (texture= bassa) & (mediaResp > -15641)) allora numero_cluster= 4 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= cartoon) & (-208974 < mediaResp ≤ -11852)) allora numero_cluster= 4 (3.0/1.1)

➤ *Caso: Clusters determinati con gerarchico e $k=6$*

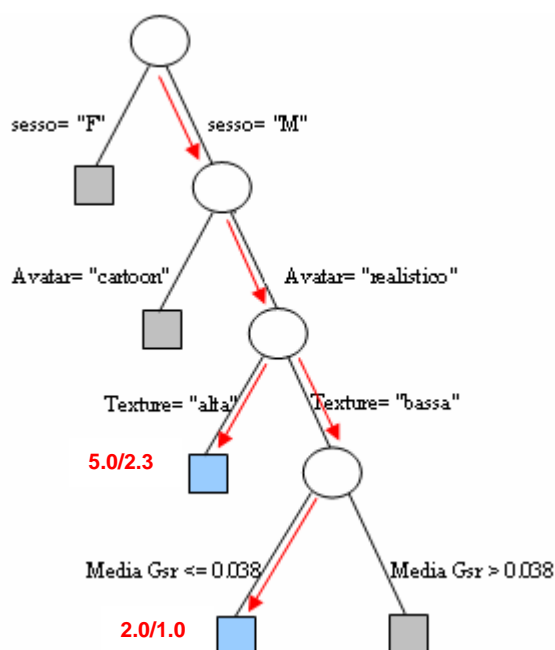
• **Cluster 4**

- Se ((sesso= maschio) & (texture= alta) & (mediaGSR > 0.019)) allora numero_cluster= 4 (4.0/2.2)
- Se ((sesso= maschio) & (texture= bassa) & (avatar= realistico)) allora numero_cluster= 4 (5.0/1.2)

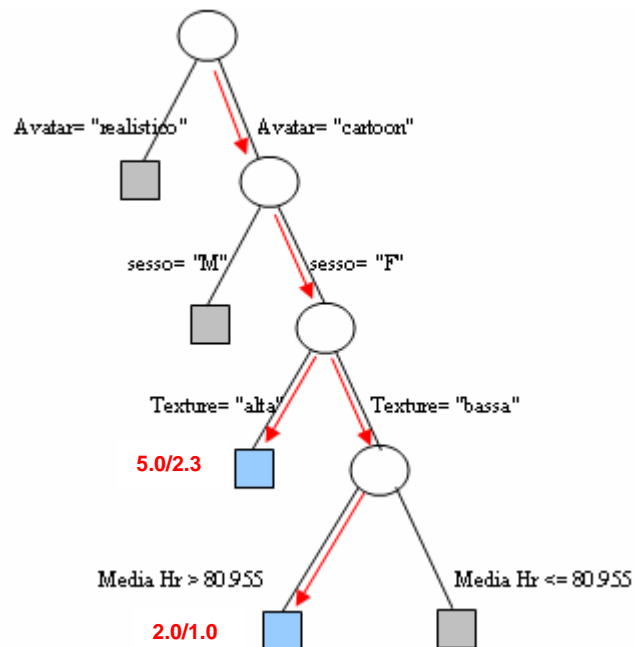
AC)

➤ *Caso: Clusters determinati con k -means e $k=5$*

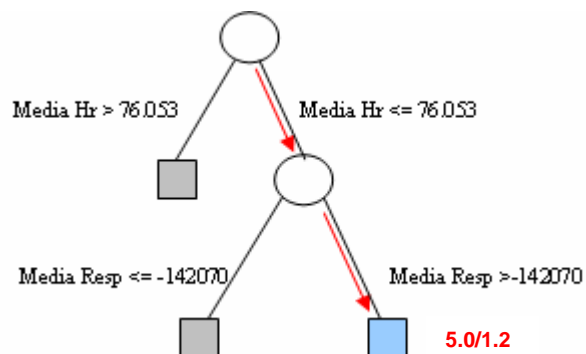
• **Cluster 0**



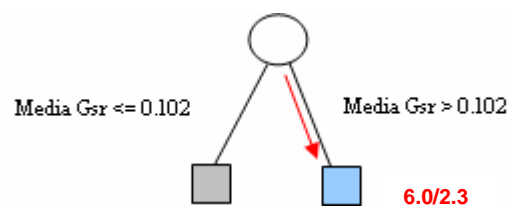
- Cluster 2



- Cluster 3

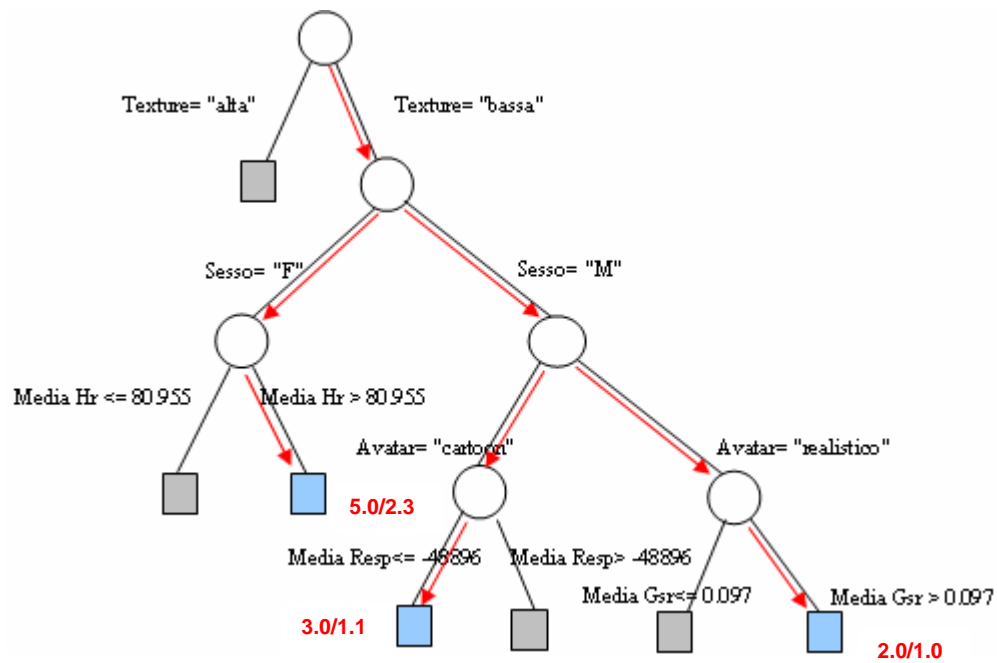


- Cluster 4

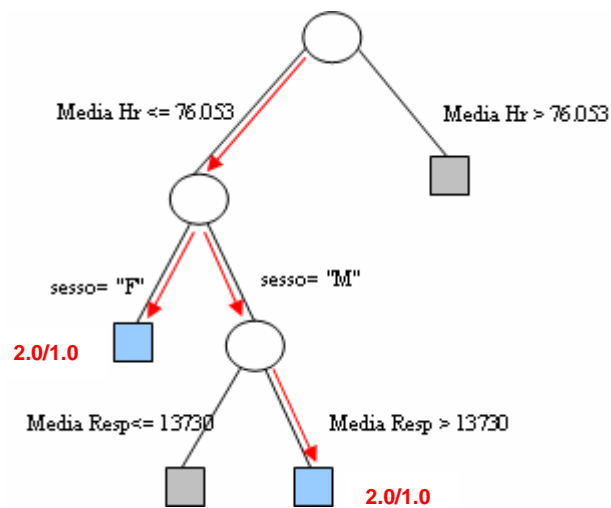


➤ *Caso: Clusters determinati con k-means e k= 6*

• **Cluster 4**



• **Cluster 5**



➤ *Caso: Clusters determinati con gerarchico e k= 5*

• **Cluster 2**

- Se ((texture= alta) & (mediaGSR <= -0.043)) allora numero_cluster= 2 (4.0/2.2)
- Se ((texture= bassa) & (mediaHR <= 80.95) & (sesso= femmina)) allora numero_cluster= 2 (5.0/2.3)

• **Cluster 3**

- Se ((sesso= maschio) & (texture= alta) & (mediaGSR > 0.019)) allora numero_cluster= 3 (4.0/2.2)
- Se ((sesso= maschio) & (texture= bassa) & (avatar= cartoon) & (mediaHR > 80.95)) allora numero_cluster= 3 (3.0/2.1)
- Se ((sesso= maschio) & (texture= bassa) & (avatar= realistico)) allora numero_cluster= 3 (5.0/1.2)

• **Cluster 4**

- Se ((sesso= femmina) & (texture= alta) & (avatar= cartoon)) allora numero_cluster= 4 (5.0/2.3)
- Se ((sesso= femmina) & (texture= alta) & (avatar= realistico) & (mediaGSR <= 0.014)) allora numero_cluster= 4 (3.0/2.1)
- Se ((sesso= femmina) & (texture= bassa) & (mediaResp <= -220151)) allora numero_cluster= 4 (2.0/1.0)
- Se ((sesso= femmina) & (texture= bassa) & (mediaResp > -15641)) allora numero_cluster= 4 (3.0/2.1)
- Se ((sesso= maschio) & (avatar= cartoon) & (-208974 < mediaResp <= -11852)) allora numero_cluster= 4 (3.0/1.1)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

• **Cluster 3**

- Se ((texture= alta) & (mediaGSR <= -0.043)) allora numero_cluster= 3 (4.0/2.2)
- Se ((texture= bassa) & (mediaHR <= 80.95) & (sesso= femmina)) allora numero_cluster= 3 (5.0/2.3)

• **Cluster 4**


- Se ((sesso= maschio) & (texture= alta) & (mediaGSR > 0.019)) allora numero_cluster= 4 (4.0/2.2)
- Se ((sesso= maschio) & (texture= bassa) & (avatar= cartoon) & (mediaHR > 80.95)) allora numero_cluster= 4 (3.0/2.1)
- Se ((sesso= maschio) & (texture= bassa) & (avatar= realistico)) allora numero_cluster= 4 (5.0/1.2)


Nella tabella B-3 sono indicati gli attributi che sono stati considerati.

	A	B	C	D	E	F	G	H	I	L	M	N	O	P	Q	R	S	T	U	V	Z	AA
<i>clustering</i>																						
<i>Bips</i>	<i>Valori</i>																					
	<i>Int.regolari</i>																					
	<i>Int.irregolari</i>																					
<i>Media GSR</i>																						
<i>Media RESP</i>																						
<i>Media Heart Rate</i>																						
<i>Texture</i>																						
<i>Avatar</i>																						
<i>Sesso</i>																						
<i>Altri Dati(*)</i>																						

Tabella B-2 Attributi utilizzati nelle varie classificazioni

Legenda



 *target*

 *attributo*

(*) *AltriDati: Occupy/ Medicine/ Language/ Alcohol/ Literate/ Programming/ VReality/ Games/ Gametimes*

	AB	AC
<i>Cluster n (n= 0,...,4)(n= 0,...5)</i>		
<i>Media GSR</i>		
<i>Media RESP</i>		
<i>Media Heart Rate</i>		
<i>Texture</i>		
<i>Avatar</i>		
<i>Sesso</i>		

Tabella B-3 Approccio ClusterNotCluster: attributi considerati

<i>Legenda</i>	
	<i>target</i>
	<i>attributo</i>

Appendice C

Le ottimizzazioni previste nel paragrafo 6.8 hanno consentito, in fase di classificazione, l'estrazione di regole di decisione interessanti. L'output dell'algoritmo di classificazione è diverso a seconda degli attributi che si considerano. In questa fase dell'analisi abbiamo considerato l'attributo cluster determinato con l'algoritmo gerarchico. Di seguito sono riportate tutte le regole estratte durante il processo di classificazione. Ogni classificazione ha coinvolto alcuni o tutti gli attributi: nella tabella C-1, per ogni classificazione sono indicati gli attributi considerati.

A)

➤ *Caso: Clusters determinati con gerarchico e $k=5$*

- numero_cluster= 4 (40.0/28.5)

➤ *Caso: Clusters determinati con gerarchico e $k=6$*

- Se (numero_BIPs = 0) allora numero_cluster= 5 (5.0/4.1)
- Se (numero_BIPs = 1) allora numero_cluster= 0 (3.0/2.8)
- Se (numero_BIPs = 2) allora numero_cluster= 4 (8.0/6.3)
- Se (numero_BIPs = 3) allora numero_cluster= 4 (7.0/4.4)
- Se (numero_BIPs = 4) allora numero_cluster= 3 (4.0/3.1)
- Se (numero_BIPs = 5) allora numero_cluster= 3 (6.0/4.3)
- Se (numero_BIPs = 6) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs = 8) allora numero_cluster= 4 (2.0/1.0)
- Se (numero_BIPs = 10) allora numero_cluster= 4 (1.0/0.8)
- Se (numero_BIPs = 11) allora numero_cluster= 5 (2.0/1.0)
- Se (numero_BIPs = 16) allora numero_cluster= 0 (1.0/0.8)

B)

➤ *Caso: Clusters determinati con gerarchico e $k=5$*

- Se ((VariazGSR <= 0.616) & (avatar= cartoon)) allora numero_cluster= 2 (3.0/1.0)

- Se ((avatar= realistico) & (texture= alta) & (VariazGSR <= 5.515)) allora numero_cluster= 0
(2.0/1.0)
- Se ((-5.515 < VariazGSR <= 0.616) & (avatar= realistico) & (texture= alta)) allora numero_cluster= 0
(2.0/1.0)
- Se ((VariazGSR <= 0.616) & (avatar= realistico) & (texture= bassa)) allora numero_cluster= 3
(2.0)
- Se ((0.616 < VariazGSR <= 7.008) & (sesso= femmina) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 4
(3.0)
- Se ((0.616 < VariazGSR <= 7.008) & (sesso= femmina) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 2
(3.0/1.0)
- Se ((0.616 < VariazGSR <= 2.3) & (sesso= femmina) & (avatar= realistico) & (VariazHR <= -0.3)) allora numero_cluster= 1
(2.0/1.0)
- Se ((2.3 < VariazGSR <= 7.008) & (sesso= femmina) & (avatar= realistico) & (VariazHR <= -0.3)) allora numero_cluster= 4
(2.0)
- Se ((0.616 < VariazGSR <= 7.008) & (sesso= femmina) & (avatar= realistico) & (VariazHR > -0.3)) allora numero_cluster= 2
(3.0/1.0)
- Se ((7.008 < VariazGSR <= 8.304) & (sesso= femmina)) allora numero_cluster= 3
(2.0)
- Se ((VariazGSR > 8.304) & (sesso = femmina)) allora numero_cluster= 4
(3.0)
- Se ((VariazGSR > 0.616) & (sesso= maschio) & (avatar= cartoon) & (VariazHR <= -0.15) & (texture= alta)) allora numero_cluster= 4
(2.0)
- Se ((VariazGSR > 0.616) & (sesso= maschio) & (avatar= cartoon) & (VariazHR <= -0.15) & (texture= bassa)) allora numero_cluster= 3
(4.0/2.0)
- Se ((VariazGSR > 0.616) & (sesso= maschio) & (avatar= cartoon) & (VariazHR > -0.15)) allora numero_cluster= 1
(2.0/1.0)
- Se ((VariazGSR > 0.616) & (sesso= maschio) & (avatar= realistico) & (texture= alta)) allora numero_cluster= 2
(2.0/1.0)
- Se ((VariazGSR > 0.616) & (sesso= maschio) & (avatar= realistico) & (texture= bassa)) allora numero_cluster= 3
(3.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((VariazGSR <= 0.616) & (avatar= cartoon)) allora numero_cluster= 3
(3.0/1.0)
- Se ((avatar= realistico) & (texture= alta) & (VariazGSR <= 5.515)) allora numero_cluster= 0
(2.0/1.0)
- Se ((-5.515 < VariazGSR <= 0.616) & (avatar= realistico) & (texture= alta)) allora numero_cluster= 0
(2.0/1.0)
- Se ((VariazGSR <= 0.616) & (avatar= realistico) & (texture= bassa)) allora numero_cluster= 4
(2.0)

- Se $((0.616 < \text{VariazGSR} \leq 7.008) \& (\text{sezzo} = \text{femmina}) \& (\text{avatar} = \text{cartoon}) \& (\text{texture} = \text{alta}))$ allora
numero_cluster= 5 (3.0/1.0)
- Se $((0.616 < \text{VariazGSR} \leq 7.008) \& (\text{sezzo} = \text{femmina}) \& (\text{avatar} = \text{cartoon}) \& (\text{texture} = \text{bassa}))$ allora
numero_cluster= 3 (3.0/1.0)
- Se $((0.616 < \text{VariazGSR} \leq 7.008) \& (\text{sezzo} = \text{femmina}) \& (\text{avatar} = \text{realistico}) \& (\text{VariazHR} \leq -0.96))$
allora numero_cluster= 5 (2.0)
- Se $((0.616 < \text{VariazGSR} \leq 2.225) \& (\text{sezzo} = \text{femmina}) \& (\text{avatar} = \text{realistico}) \& (\text{VariazHR} > -0.96))$
allora numero_cluster= 3 (2.0)
- Se $((2.225 < \text{VariazGSR} \leq 7.008) \& (\text{sezzo} = \text{femmina}) \& (\text{avatar} = \text{realistico}) \& (\text{VariazHR} > -0.96))$
allora numero_cluster= 2 (3.0/1.0)
- Se $((7.008 < \text{VariazGSR} \leq 8.304) \& (\text{sezzo} = \text{femmina}))$ allora numero_cluster= 4 (2.0)
- Se $((\text{VariazGSR} > 8.304) \& (\text{sezzo} = \text{femmina}))$ allora numero_cluster= 5 (3.0/1.0)
- Se $((0.616 < \text{VariazGSR} \leq 2.162) \& (\text{Sesso} = \text{maschio}) \& (\text{Texture} = \text{alta}))$ allora numero_cluster= 3
(3.0/1.0)
- Se $((\text{VariazGsr} > 2.162) \& (\text{Sesso} = \text{maschio}) \& (\text{Texture} = \text{alta}))$ allora numero_cluster= 2
(3.0/2.0)
- Se $((\text{VariazGsr} > 0.616) \& (\text{Sesso} = \text{maschio}) \& (\text{Texture} = \text{bassa}) \& (\text{Avatar} = \text{cartoon}))$ allora
numero_cluster= 4 (4.0/2.0)
- Se $((\text{VariazGsr} > 0.616) \& (\text{Sesso} = \text{maschio}) \& (\text{Texture} = \text{bassa}) \& (\text{Avatar} = \text{realistico}))$ allora
numero_cluster= 4 (3.0)

C)

➤ *Caso: Clusters determinati con gerarchico e $k=5$*

- Se $((\text{numeroDiBips} = 0) \& (\text{VariazGsr} \leq 1.028))$ allora numero_cluster= 4 (2.0)
- Se $((\text{numeroDiBips} = 0) \& (\text{VariazGsr} > 1.028))$ allora numero_cluster= 1 (3.0/2.0)
- Se $(\text{numeroDiBips} = 1)$ allora numero_cluster= 0 (3.0/2.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{alta}))$ allora numero_cluster= 4 (2.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{bassa}) \& (\text{VariazGsr} \leq 2.22))$ allora numero_cluster= 2
(3.0/1.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{bassa}) \& (\text{VariazGsr} > 2.22))$ allora numero_cluster= 3
(3.0/1.0)
- Se $((\text{numeroDiBips} = 3) \& (\text{Sesso} = \text{femmina}))$ allora numero_cluster= 4 (3.0/1.0)
- Se $((\text{numeroDiBips} = 3) \& (\text{Sesso} = \text{maschio}) \& (\text{VariazGsr} \leq 0.616))$ allora numero_cluster= 3
(2.0)
- Se $((\text{numeroDiBips} = 3) \& (\text{Sesso} = \text{maschio}) \& (\text{VariazGsr} > 0.616))$ allora numero_cluster= 3
(2.0/1.0)
- Se $((\text{numeroDiBips} = 4) \& (\text{VariazGsr} \leq 9.702))$ allora numero_cluster= 2 (2.0)
- Se $((\text{numeroDiBips} = 4) \& (\text{VariazGsr} > 9.702))$ allora numero_cluster= 4 (2.0)

- Se ((numeroDiBips= 5) & (Texture= alta) & (VariazGsr <= 1.295)) allora numero_cluster= 2
(2.0/1.0)
- Se ((numeroDiBips= 5) & (Texture= alta) & (VariazGsr > 1.295)) allora numero_cluster= 4
(2.0)
- Se ((numeroDiBips= 5) & (Texture= bassa)) allora numero_cluster= 2
(2.0)
- Se (numeroDiBips= 6) allora numero_cluster= 3
(1.0)
- Se (numeroDiBips= 8) allora numero_cluster= 3
(2.0)
- Se (numeroDiBips= 10) allora numero_cluster= 3
(1.0)
- Se (numeroDiBips = 11) allora numero_cluster= 4
(2.0)
- Se (numeroDiBips= 16) allora numero_cluster= 0
(1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((numeroDiBips= 0) & (VariazGsr <= 1.028)) allora numero_cluster= 5
(2.0)
- Se ((numeroDiBips= 0) & (VariazGsr > 1.028)) allora numero_cluster= 2
(3.0/2.0)
- Se (numeroDiBips= 1) allora numero_cluster= 0
(3.0/2.0)
- Se ((numeroDiBips= 2) & (Texture= alta)) allora numero_cluster= 5
(2.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazGsr <= 2.225)) allora numero_cluster= 3
(3.0/1.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazGsr > 2.225)) allora numero_cluster= 4
(3.0/1.0)
- Se ((numeroDiBips= 3) & (Avatar= cartoon) & (Sesso= femmina)) allora numero_cluster= 4
(2.0/1.0)
- Se ((numeroDiBips= 3) & (Avatar= cartoon) & (Sesso= maschio)) allora numero_cluster= 4
(2.0/1.0)
- Se ((numeroDiBips= 3) & (Avatar= realistico)) allora numero_cluster= 4
(3.0/1.0)
- Se ((numeroDiBips= 4) & (VariazGsr <= 9.702)) allora numero_cluster= 3
(2.0)
- Se ((numeroDiBips= 4) & (VariazGsr > 9.702)) allora numero_cluster= 1
(2.0/1.0)
- Se ((numeroDiBips= 5) & (Texture= alta) & (VariazGsr <= 1.295)) allora numero_cluster= 1
(2.0/1.0)
- Se ((numeroDiBips= 5) & (Texture= alta) & (VariazGsr > 1.295)) allora numero_cluster= 1
(2.0/1.0)
- Se ((numeroDiBips= 5) & (Texture= bassa)) allora numero_cluster= 3
(2.0)
- Se (numeroDiBips= 6) allora numero_cluster= 4
(1.0)
- Se (numeroDiBips= 8) allora numero_cluster= 4
(2.0)
- Se (numeroDiBips= 10) allora numero_cluster= 4
(1.0)
- Se (numeroDiBips= 11) allora numero_cluster= 5
(2.0)
- Se (numeroDiBips= 16) allora numero_cluster= 0
(1.0)

D)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((numeroDiBips= 0) & (VariazGsr <= 1.03)) allora numero_cluster= 4 (2.0)
- Se ((numeroDiBips= 0) & (VariazGsr > 1.03)) allora numero_cluster= 1 (3.0/2.0)
- Se (numeroDiBips= 1) allora numero_cluster= 0 (3.0/2.0)
- Se ((numeroDiBips= 2) & (Texture= alta)) allora numero_cluster= 4 (2.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazHr <= -0.53)) allora numero_cluster= 3 (3.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazHr > -0.53)) allora numero_cluster= 2 (3.0/1.0)
- Se ((numeroDiBips= 3) & (VariazHr <= -0.96)) allora numero_cluster= 4 (2.0)
- Se ((numeroDiBips= 3) & (VariazHr > -0.96) & (Sesso= femmina)) allora numero_cluster= 3 (2.0/1.0)
- Se ((numeroDiBips= 3) & (VariazHr > -0.96) & (Sesso= maschio)) allora numero_cluster= 3 (3.0)
- Se ((numeroDiBips= 4) & (VariazGsr <= 9.702)) allora numero_cluster= 2 (2.0)
- Se ((numeroDiBips= 4) & (VariazGsr > 9.702)) allora numero_cluster= 4 (2.0)
- Se ((numeroDiBips= 5) & (Texture= alta) & (VariazGsr <= 1.295)) allora numero_cluster= 2 (2.0/1.0)
- Se ((numeroDiBips= 5) & (Texture= alta) & (VariazGsr > 1.295)) allora numero_cluster= 4 (2.0)
- Se ((numeroDiBips= 5) & (Texture= bassa)) allora numero_cluster= 2 (2.0)
- Se (numeroDiBips= 6) allora numero_cluster= 3 (1.0)
- Se (numeroDiBips= 8) allora numero_cluster= 3 (2.0)
- Se (numeroDiBips= 10) allora numero_cluster= 3 (1.0)
- Se (numeroDiBips= 11) allora numero_cluster= 4 (2.0)
- Se (numeroDiBips= 16) allora numero_cluster= 0 (1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((numeroDiBips= 0) & (VariazGsr <= 1.03)) allora numero_cluster= 5 (2.0)
- Se ((numeroDiBips= 0) & (VariazGsr > 1.03)) allora numero_cluster= 2 (3.0/2.0)
- Se (numeroDiBips= 1) allora numero_cluster= 0 (3.0/2.0)
- Se ((numeroDiBips= 2) & (Texture= alta)) allora numero_cluster= 5 (2.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazHr <= -0.53)) allora numero_cluster= 4 (3.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazHr > -0.53)) allora numero_cluster= 3 (3.0/1.0)
- Se ((numeroDiBips= 3) & (VariazHr <= -0.96)) allora numero_cluster= 5 (2.0)

- Se ((numeroDiBips= 3) & (VariazHr > -0.96) & (Sesso= femmina)) allora numero_cluster= 1
(2.0/1.0)
- Se ((numeroDiBips= 3) & (VariazHr > -0.96) & (Sesso= maschio)) allora numero_cluster= 4
(3.0)
- Se ((numeroDiBips= 4) & (VariazGsr <= 9.702)) allora numero_cluster= 3
(2.0)
- Se ((numeroDiBips= 4) & (VariazGsr > 9.702)) allora numero_cluster= 1
(2.0/1.0)
- Se ((numeroDiBips= 5) & (Texture= alta) & (VariazHr <= 0.12)) allora numero_cluster= 1
(2.0)
- Se ((numeroDiBips= 5) & (Texture= alta) & (VariazHr > 0.12)) allora numero_cluster= 3
(2.0/1.0)
- Se ((numeroDiBips= 5) & (Texture= bassa)) allora numero_cluster= 3
(2.0)
- Se (numeroDiBips= 6) allora numero_cluster= 4
(1.0)
- Se (numeroDiBips= 8) allora numero_cluster= 4
(2.0)
- Se (numeroDiBips= 10) allora numero_cluster= 4
(1.0)
- Se (numeroDiBips= 11) allora numero_cluster= 5
(2.0)
- Se (numeroDiBips= 16) allora numero_cluster= 0
(1.0)

E)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((0 <= numeroDiBips <= 3) & (Sesso= femmina) & (VariazGsr <= 7.01) & (Texture= alta)) allora numero_cluster= 4
(4.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= femmina) & (VariazGsr <= 2.225) & (Texture= bassa)) allora numero_cluster= 2
(3.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= femmina) & (2.225 < VariazGsr <= 7.01) & (Texture= bassa)) allora numero_cluster= 1
(2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= femmina) & (VariazGsr > 7.01)) allora numero_cluster= 3
(2.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= maschio) & (Avatar= cartoon) & (Texture= alta)) allora numero_cluster= 1
(3.0/2.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= maschio) & (Avatar= cartoon) & (Texture= bassa)) allora numero_cluster= 2
(3.0/2.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= maschio) & (Avatar= realistico) & (Texture= alta)) allora numero_cluster= 0
(3.0/2.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= maschio) & (Avatar= realistico) & (Texture= bassa)) allora numero_cluster= 3
(3.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= cartoon)) allora numero_cluster= 4
(3.0/1.0)

- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= realistico)) allora numero_cluster= 2 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= bassa)) allora numero_cluster= 2 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (4.99 < VariazGsr <= 24.99)) allora numero_cluster= 4 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr > 24.99)) allora numero_cluster= 3 (2.0/1.0)
- Se ((8 <= numeroDiBips <= 11) & (VariazGsr <= 1.29)) allora numero_cluster= 3 (2.0)
- Se ((8 <= numeroDiBips <= 11) & (VariazGsr > 1.29)) allora numero_cluster= 4 (3.0/1.0)
- Se (numeroDiBips= 16) allora numero_cluster= 0 (1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= alta) & (VariazGsr <= 2.3)) allora numero_cluster= 2 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= alta) & (2.3 < VariazGsr <= 4.99)) allora numero_cluster= 5 (2.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= alta) & (VariazGsr > 4.99)) allora numero_cluster= 4 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= bassa) & (VariazGsr <= 1.028)) allora numero_cluster= 5 (3.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= bassa) & (VariazGsr > 1.028)) allora numero_cluster= 3 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= femmina) & (VariazGsr <= 2.225)) allora numero_cluster= 3 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= femmina) & (2.225 < VariazGsr <= 4.99)) allora numero_cluster= 1 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= femmina) & (VariazGsr > 4.99)) allora numero_cluster= 2 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= maschio) & (Texture= alta)) allora numero_cluster= 0 (3.0/2.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= maschio) & (Texture= bassa)) allora numero_cluster= 4 (3.0)
- Se ((4<= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= cartoon)) allora numero_cluster= 1 (3.0/1.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= realistico)) allora numero_cluster= 3 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= bassa)) allora numero_cluster= 3 (2.0)

- Se ((4 <= numeroDiBips <= 7) & (4.99 < VariazGsr <= 24.99)) allora numero_cluster= 1 (2.0/1.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr > 24.99)) allora numero_cluster= 4 (2.0/1.0)
- Se ((8 <= numeroDiBips <= 11) & (VariazGsr <= 1.295)) allora numero_cluster= 4 (2.0)
- Se ((8 <= numeroDiBips <= 11) & (VariazGsr > 1.295)) allora numero_cluster= 5 (3.0/1.0)
- Se (numeroDiBips= 16) allora numero_cluster= 0 (1.0)

F)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((0 <= numeroDiBips <= 3) & (Sesso= femmina) & (VariazGsr <= 7.008) & (Texture= alta)) allora numero_cluster= 4 (4.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= femmina) & (VariazGsr <= 2.225) & (Texture= bassa)) allora numero_cluster= 2 (3.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= femmina) & (2.225 < VariazGsr <= 7.008) & (texture= bassa)) allora numero_cluster= 1 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= femmina) & (VariazGsr > 7.008)) allora numero_cluster= 3 (2.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= maschio) & (VariazHr <= -0.18) & (Avatar= cartoon)) allora numero_cluster= 4 (3.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= maschio) & (VariazHr <= -0.18) & (Avatar= realistico)) allora numero_cluster= 3 (4.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= maschio) & (VariazHr > -0.18) & (Avatar= cartoon)) allora numero_cluster= 1 (3.0/2.0)
- Se ((0 <= numeroDiBips <= 3) & (Sesso= maschio) & (VariazHr > -0.18) & (Avatar= realistico)) allora numero_cluster= 0 (2.0/1.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= cartoon)) allora numero_cluster= 4 (3.0/1.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= realistico)) allora numero_cluster= 2 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= bassa)) allora numero_cluster= 2 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (4.99 < VariazGsr <= 24.995)) allora numero_cluster= 4 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr > 24.995)) allora numero_cluster= 3 (2.0/1.0)
- Se ((8 <= numeroDiBips <= 11) & (VariazGsr <= 1.295)) allora numero_cluster= 3 (2.0)
- Se ((8 <= numeroDiBips <= 11) & (VariazGsr > 1.295)) allora numero_cluster= 4 (3.0/1.0)
- Se (numeroDiBips= 16) allora numero_cluster= 0 (1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= alta) & (VariazGsr <= 2.3)) allora numero_cluster= 2 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= alta) & (2.3 < VariazGsr <= 4.99)) allora numero_cluster= 5 (2.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= alta) & (VariazGsr > 4.99)) allora numero_cluster= 4 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= bassa) & (VariazGsr <= 1.028)) allora numero_cluster= 5 (3.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= cartoon) & (Texture= bassa) & (VariazGsr > 1.028)) allora numero_cluster= 3 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= femmina) & (VariazGsr <= 2.225)) allora numero_cluster= 3 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= femmina) & (2.225 < VariazGsr <= 4.99)) allora numero_cluster= 1 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= femmina) & (VariazGsr > 4.99)) allora numero_cluster= 2 (2.0/1.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= maschio) & (VariazHr <= -0.209)) allora numero_cluster= 4 (4.0)
- Se ((0 <= numeroDiBips <= 3) & (Avatar= realistico) & (Sesso= maschio) & (VariazHr > -0.209)) allora numero_cluster= 0 (2.0/1.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= cartoon)) allora numero_cluster= 1 (3.0/1.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= realistico)) allora numero_cluster= 3 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr <= 4.99) & (Texture= bassa)) allora numero_cluster= 3 (2.0)
- Se ((4 <= numeroDiBips <= 7) & (4.99 < VariazGsr <= 24.995)) allora numero_cluster= 1 (2.0/1.0)
- Se ((4 <= numeroDiBips <= 7) & (VariazGsr > 24.995)) allora numero_cluster= 4 (2.0/1.0)
- Se ((8 <= numeroDiBips <= 11) & (VariazGsr <= 1.295)) allora numero_cluster= 4 (2.0)
- Se ((8 <= numeroDiBips <= 11) & (VariazGsr > 1.295)) allora numero_cluster= 5 (3.0/1.0)
- Se (numeroDiBips= 16) allora numero_cluster= 0 (1.0)

G)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((0 <= numeroDiBips <= 1) & (VariazGsr <= 1.028)) allora numero_cluster= 4 (3.0/1.0)

- Se $((0 \leq \text{numeroDiBips} \leq 1) \& (1.028 < \text{VariazGsr} \leq 1.295))$ allora $\text{numero_cluster} = 2$
(2.0)
- Se $((0 \leq \text{numeroDiBips} \leq 1) \& (\text{VariazGsr} > 1.295))$ allora $\text{numero_cluster} = 1$
(3.0/1.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{alta}))$ allora $\text{numero_cluster} = 4$
(2.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{bassa}) \& (\text{VariazGsr} \leq 2.225))$ allora $\text{numero_cluster} = 2$
(3.0/1.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{bassa}) \& (\text{VariazGsr} > 2.225))$ allora $\text{numero_cluster} = 3$
(3.0/1.0)
- Se $((\text{numeroDiBips} = 3) \& (\text{Sesso} = \text{femmina}))$ allora $\text{numero_cluster} = 4$
(3.0/1.0)
- Se $((\text{numeroDiBips} = 3) \& (\text{Sesso} = \text{maschio}) \& (\text{VariazGsr} \leq 0.616))$ allora $\text{numero_cluster} = 3$
(2.0)
- Se $((\text{numeroDiBips} = 3) \& (\text{Sesso} = \text{maschio}) \& (\text{VariazGsr} > 0.616))$ allora $\text{numero_cluster} = 3$
(2.0/1.0)
- Se $((4 \leq \text{numeroDiBips} \leq 5) \& (\text{VariazGsr} \leq 0.844))$ allora $\text{numero_cluster} = 2$
(3.0)
- Se $((4 \leq \text{numeroDiBips} \leq 5) \& (\text{VariazGsr} > 0.844) \& (\text{Texture} = \text{alta}) \& (\text{Avatar} = \text{cartoon}))$ allora
 $\text{numero_cluster} = 4$
(3.0)
- Se $((4 \leq \text{numeroDiBips} \leq 5) \& (\text{VariazGsr} > 0.844) \& (\text{Texture} = \text{alta}) \& (\text{Avatar} = \text{realistico}))$ allora
 $\text{numero_cluster} = 2$
(2.0/1.0)
- Se $((4 \leq \text{numeroDiBips} \leq 5) \& (\text{VariazGsr} > 0.844) \& (\text{Texture} = \text{bassa}))$ allora $\text{numero_cluster} = 2$
(2.0/1.0)
- Se $((6 \leq \text{numeroDiBips} \leq 16) \& (\text{Sesso} = \text{femmina}))$ allora $\text{numero_cluster} = 0$
(2.0/1.0)
- Se $((6 \leq \text{numeroDiBips} \leq 16) \& (\text{Sesso} = \text{maschio}) \& (\text{Avatar} = \text{cartoon}))$ allora $\text{numero_cluster} = 3$
(2.0/1.0)
- Se $((6 \leq \text{numeroDiBips} \leq 16) \& (\text{Sesso} = \text{maschio}) \& (\text{Avatar} = \text{realistico}))$ allora $\text{numero_cluster} = 3$
(3.0)

➤ *Caso: Clusters determinati con gerarchico e $k=6$*

- Se $((0 \leq \text{numeroDiBips} \leq 1) \& (\text{VariazGsr} \leq 1.028))$ allora $\text{numero_cluster} = 5$
(3.0/1.0)
- Se $((0 \leq \text{numeroDiBips} \leq 1) \& (1.028 < \text{VariazGsr} \leq 1.295))$ allora $\text{numero_cluster} = 3$
(2.0)
- Se $((0 \leq \text{numeroDiBips} \leq 1) \& (\text{VariazGsr} > 1.295))$ allora $\text{numero_cluster} = 2$
(3.0/1.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{alta}))$ allora $\text{numero_cluster} = 5$
(2.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{bassa}) \& (\text{VariazGsr} \leq 2.225))$ allora $\text{numero_cluster} = 3$
(3.0/1.0)
- Se $((\text{numeroDiBips} = 2) \& (\text{Texture} = \text{bassa}) \& (\text{VariazGsr} > 2.225))$ allora $\text{numero_cluster} = 4$
(3.0/1.0)
- Se $((\text{numeroDiBips} = 3) \& (\text{Avatar} = \text{cartoon}) \& (\text{Sesso} = \text{femmina}))$ allora $\text{numero_cluster} = 4$
(2.0/1.0)
- Se $((\text{numeroDiBips} = 3) \& (\text{Avatar} = \text{cartoon}) \& (\text{Sesso} = \text{maschio}))$ allora $\text{numero_cluster} = 4$
(2.0/1.0)

- Se ((numeroDiBips= 3) & (Avatar= realistico)) allora numero_cluster= 4 (3.0/1.0)
- Se ((4<=numeroDiBips<=5) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= cartoon)) allora numero_cluster= 1 (3.0/1.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= realistico)) allora numero_cluster= 3 (2.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr <= 4.99) & (Texture= bassa)) allora numero_cluster= 3 (2.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr > 4.99)) allora numero_cluster= 5 (3.0/1.0)
- Se ((6 <= numeroDiBips <= 16) & (Sesso= femmina)) allora numero_cluster= 0 (2.0/1.0)
- Se ((6 <= numeroDiBips <= 16) & (Sesso= maschio) & (Avatar= cartoon)) allora numero_cluster= 4 (2.0/1.0)
- Se ((6 <= numeroDiBips <= 16) & (Sesso= maschio) & (Avatar= realistico)) allora numero_cluster= 4 (3.0)

H)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((0 <= numeroDiBips <= 1) & (VariazGsr <= 1.028)) allora numero_cluster= 4 (3.0/1.0)
- Se ((0 <= numeroDiBips <= 1) & (1.028 < VariazGsr <= 1.295)) allora numero_cluster= 2 (2.0)
- Se ((0 <= numeroDiBips <= 1) & (VariazGsr > 1.295)) allora numero_cluster= 1 (3.0/1.0)
- Se ((numeroDiBips= 2) & (Texture= alta)) allora numero_cluster= 4 (2.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazHr <= -0.53)) allora numero_cluster= 3 (3.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazHr > -0.53)) allora numero_cluster= 2 (3.0/1.0)
- Se ((numeroDiBips= 3) & (VariazHr <= -0.96)) allora numero_cluster= 4 (2.0)
- Se ((numeroDiBips= 3) & (VariazHr > -0.96) & (Sesso= femmina)) allora numero_cluster= 3 (2.0/1.0)
- Se ((numeroDiBips= 3) & (VariazHr > -0.96) & (Sesso= maschio)) allora numero_cluster= 3 (3.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr <= 0.844)) allora numero_cluster= 2 (3.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr > 0.844) & (VariazHr <= -0.209)) allora numero_cluster= 4 (4.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr > 0.844) & (VariazHr > -0.209)) allora numero_cluster= 2 (3.0/1.0)
- Se ((6 <= numeroDiBips <= 16) & (Sesso= femmina)) allora numero_cluster= 0 (2.0/1.0)
- Se ((6 <= numeroDiBips <= 16) & (Sesso= maschio) & (Avatar= cartoon)) allora numero_cluster= 3 (2.0/1.0)

- Se ((6 <= numeroDiBips <= 16) & (Sesso= maschio) & (Avatar= realistico)) allora numero_cluster= 3
(3.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((0 <= numeroDiBips <= 1) & (VariazGsr <= 1.028)) allora numero_cluster= 5 (3.0/1.0)
- Se ((0 <= numeroDiBips <= 1) & (1.028 < VariazGsr <= 1.536)) allora numero_cluster= 3
(2.0)
- Se ((0 <= numeroDiBips <= 1) & (VariazGsr > 1.536)) allora numero_cluster= 2 (3.0/1.0)
- Se ((numeroDiBips= 2) & (Texture= alta)) allora numero_cluster= 5 (2.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazHr <= -0.53)) allora numero_cluster= 4
(3.0)
- Se ((numeroDiBips= 2) & (Texture= bassa) & (VariazHr > -0.53)) allora numero_cluster= 3
(3.0/1.0)
- Se ((numeroDiBips= 3) & (VariazHr <= -0.96)) allora numero_cluster= 5 (2.0)
- Se ((numeroDiBips= 3) & (VariazHr > -0.96) & (Sesso= femmina)) allora numero_cluster= 1
(2.0/1.0)
- Se ((numeroDiBips= 3) & (VariazHr > -0.96) & (Sesso= maschio)) allora numero_cluster= 4
(3.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= cartoon)) allora numero_cluster= 1 (3.0/1.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr <= 4.99) & (Texture= alta) & (Avatar= realistico)) allora numero_cluster= 3 (2.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr <= 4.99) & (Texture= bassa)) allora numero_cluster= 3
(2.0)
- Se ((4 <= numeroDiBips <= 5) & (VariazGsr > 4.99)) allora numero_cluster= 5 (3.0/1.0)
- Se ((6 <= numeroDiBips <= 16) & (Sesso= femmina)) allora numero_cluster= 0 (2.0/1.0)
- Se ((6 <= numeroDiBips <= 16) & (Sesso= maschio) & (Avatar= cartoon)) allora numero_cluster= 4
(2.0/1.0)
- Se ((6 <= numeroDiBips <= 16) & (Sesso= maschio) & (Avatar= realistico)) allora numero_cluster= 4
(3.0)

I)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1)) allora numero_cluster= 4 (8.0/2.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2)) allora numero_cluster= 4 (4.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 3)) allora numero_cluster= 3 (1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 4)) allora numero_cluster= 2 (1.0)
- Se ((GameTimes= 1) & (Programming= 2)) allora numero_cluster= 4 (1.0)

- Se ((GameTimes= 1) & (Programming= 3)) allora numero_cluster= 3 (2.0)
- Se ((GameTimes= 1) & (Programming= 4)) allora numero_cluster= 2 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 5)) allora numero_cluster= 1 (2.0/1.0)
- Se ((GameTimes=2) & (Literate= 3)) allora numero_cluster= 1 (3.0/1.0)
- Se ((GameTimes=2) & (Literate= 4)) allora numero_cluster= 4 (3.0/1.0)
- Se ((GameTimes=2) & (Literate= 5)) allora numero_cluster= 2 (3.0)
- Se ((GameTimes= 3) & (Avatar= cartoon)) allora numero_cluster= 3 (3.0)
- Se ((GameTimes= 3) & (Avatar= realistico)) allora numero_cluster= 2 (2.0/1.0)
- Se (GameTimes= 4) allora numero_cluster= 2 (2.0/1.0)
- Se (GameTimes= 5) allora numero_cluster= 3 (1.0)
- Se (GameTimes= 6) allora numero_cluster= 0 (1.0)
- Se (GameTimes= 7) allora numero_cluster= 3 (1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (Literate= 3)) allora numero_cluster= 5 (3.0/1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (Literate= 4)) allora numero_cluster= 5 (2.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (Literate= 5)) allora numero_cluster= 0 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (Literate= 7)) allora numero_cluster= 1 (1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2) & (VariazGsr <= 3.14)) allora numero_cluster= 5 (2.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2) & (VariazGsr > 3.14)) allora numero_cluster= 1 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 3)) allora numero_cluster= 4 (1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 4)) allora numero_cluster= 3 (1.0)
- Se ((GameTimes= 1) & (Programming= 2)) allora numero_cluster= 5 (1.0)
- Se ((GameTimes= 1) & (Programming= 3)) allora numero_cluster= 4 (2.0)
- Se ((GameTimes= 1) & (Programming= 4)) allora numero_cluster= 3 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 5)) allora numero_cluster= 2 (2.0/1.0)
- Se ((GameTimes= 2) & (Literate= 3)) allora numero_cluster= 2 (3.0/1.0)
- Se ((GameTimes= 2) & (Literate= 4)) allora numero_cluster= 1 (3.0/1.0)
- Se ((GameTimes= 2) & (Literate= 5)) allora numero_cluster= 3 (3.0)
- Se ((GameTimes= 3) & (Avatar= cartoon)) allora numero_cluster= 4 (3.0)
- Se ((GameTimes= 3) & (Avatar= realistico)) allora numero_cluster= 3 (2.0/1.0)
- Se (GameTimes= 4) allora numero_cluster= 3 (2.0/1.0)

- Se (GameTimes= 5) allora numero_cluster= 4 (1.0)
- Se (GameTimes= 6) allora numero_cluster= 0 (1.0)
- Se (GameTimes= 7) allora numero_cluster= 4 (1.0)

L)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1)) allora numero_cluster= 4 (8.0/2.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2)) allora numero_cluster= 4 (4.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 3)) allora numero_cluster= 3 (1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 4)) allora numero_cluster= 2 (1.0)
- Se ((GameTimes= 1) & (Programming= 2)) allora numero_cluster= 4 (1.0)
- Se ((GameTimes= 1) & (Programming= 3)) allora numero_cluster= 3 (2.0)
- Se ((GameTimes= 1) & (Programming= 4)) allora numero_cluster= 2 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 5)) allora numero_cluster= 1 (2.0/1.0)
- Se ((GameTimes=2) & (Literate= 3)) allora numero_cluster= 1 (3.0/1.0)
- Se ((GameTimes=2) & (Literate= 4)) allora numero_cluster= 4 (3.0/1.0)
- Se ((GameTimes=2) & (Literate= 5)) allora numero_cluster= 2 (3.0)
- Se ((GameTimes= 3) & (Avatar= cartoon)) allora numero_cluster= 3 (3.0)
- Se ((GameTimes= 3) & (Avatar= realistico)) allora numero_cluster= 2 (2.0/1.0)
- Se (GameTimes= 4) allora numero_cluster= 2 (2.0/1.0)
- Se (GameTimes= 5) allora numero_cluster= 3 (1.0)
- Se (GameTimes= 6) allora numero_cluster= 0 (1.0)
- Se (GameTimes= 7) allora numero_cluster= 3 (1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (0 <= numeroDiBips <= 3)) allora numero_cluster= 5 (3.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (4 <= numeroDiBips <= 7)) allora numero_cluster= 1 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (8 <= numeroDiBips <= 11)) allora numero_cluster= 4 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (numeroDiBips = 16)) allora numero_cluster= 0 (1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2) & (VariazGsr <= 3.14)) allora numero_cluster= 5 (2.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2) & (VariazGsr > 3.14)) allora numero_cluster= 1 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 3)) allora numero_cluster= 4 (1.0)

- Se ((GameTimes= 1) & (Programming= 1) & (Games= 4)) allora numero_cluster= 3 (1.0)
- Se ((GameTimes= 1) & (Programming= 2)) allora numero_cluster= 5 (1.0)
- Se ((GameTimes= 1) & (Programming= 3)) allora numero_cluster= 4 (2.0)
- Se ((GameTimes= 1) & (Programming= 4)) allora numero_cluster= 3 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 5)) allora numero_cluster= 2 (2.0/1.0)
- Se ((GameTimes= 2) & (Literate= 3)) allora numero_cluster= 2 (3.0/1.0)
- Se ((GameTimes= 2) & (Literate= 4)) allora numero_cluster= 1 (3.0/1.0)
- Se ((GameTimes= 2) & (Literate= 5)) allora numero_cluster= 3 (3.0)
- Se ((GameTimes= 3) & (Avatar= cartoon)) allora numero_cluster= 4 (3.0)
- Se ((GameTimes= 3) & (Avatar= realistico)) allora numero_cluster= 3 (2.0/1.0)
- Se (GameTimes= 4) allora numero_cluster= 3 (2.0/1.0)
- Se (GameTimes= 5) allora numero_cluster= 4 (1.0)
- Se (GameTimes= 6) allora numero_cluster= 0 (1.0)
- Se (GameTimes= 7) allora numero_cluster= 4 (1.0)

M)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1)) allora numero_cluster= 4 (8.0/2.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2)) allora numero_cluster= 4 (4.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 3)) allora numero_cluster= 3 (1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 4)) allora numero_cluster= 2 (1.0)
- Se ((GameTimes= 1) & (Programming= 2)) allora numero_cluster= 4 (1.0)
- Se ((GameTimes= 1) & (Programming= 3)) allora numero_cluster= 3 (2.0)
- Se ((GameTimes= 1) & (Programming= 4)) allora numero_cluster= 2 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 5)) allora numero_cluster= 1 (2.0/1.0)
- Se ((GameTimes=2) & (Literate= 3)) allora numero_cluster= 1 (3.0/1.0)
- Se ((GameTimes=2) & (Literate= 4)) allora numero_cluster= 4 (3.0/1.0)
- Se ((GameTimes=2) & (Literate= 5)) allora numero_cluster= 2 (3.0)
- Se ((GameTimes= 3) & (Avatar= cartoon)) allora numero_cluster= 3 (3.0)
- Se ((GameTimes= 3) & (Avatar= realistico)) allora numero_cluster= 2 (2.0/1.0)
- Se (GameTimes= 4) allora numero_cluster= 2 (2.0/1.0)
- Se (GameTimes= 5) allora numero_cluster= 3 (1.0)
- Se (GameTimes= 6) allora numero_cluster= 0 (1.0)
- Se (GameTimes= 7) allora numero_cluster= 3 (1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (Literate= 3)) allora numero_cluster= 5 (3.0/1.0)

- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (Literate= 4)) allora numero_cluster= 5
(2.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (Literate= 5)) allora numero_cluster= 0
(2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 1) & (Literate= 7)) allora numero_cluster= 1
(1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2) & (VariazGsr <= 3.14)) allora numero_cluster= 5
(2.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 2) & (VariazGsr > 3.14)) allora numero_cluster= 1
(2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 3)) allora numero_cluster= 4 (1.0)
- Se ((GameTimes= 1) & (Programming= 1) & (Games= 4)) allora numero_cluster= 3 (1.0)
- Se ((GameTimes= 1) & (Programming= 2)) allora numero_cluster= 5 (1.0)
- Se ((GameTimes= 1) & (Programming= 3)) allora numero_cluster= 4 (2.0)
- Se ((GameTimes= 1) & (Programming= 4)) allora numero_cluster= 3 (2.0/1.0)
- Se ((GameTimes= 1) & (Programming= 5)) allora numero_cluster= 2 (2.0/1.0)
- Se ((GameTimes= 2) & (Literate= 3)) allora numero_cluster= 2 (3.0/1.0)
- Se ((GameTimes= 2) & (Literate= 4)) allora numero_cluster= 1 (3.0/1.0)
- Se ((GameTimes= 2) & (Literate= 5)) allora numero_cluster= 3 (3.0)
- Se ((GameTimes= 3) & (Avatar= cartoon)) allora numero_cluster= 4 (3.0)
- Se ((GameTimes= 3) & (Avatar= realistico)) allora numero_cluster= 3 (2.0/1.0)
- Se (GameTimes= 4) allora numero_cluster= 3 (2.0/1.0)
- Se (GameTimes= 5) allora numero_cluster= 4 (1.0)
- Se (GameTimes= 6) allora numero_cluster= 0 (1.0)
- Se (GameTimes= 7) allora numero_cluster= 4 (1.0)

N)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora numero_bips= 1 (2.0/1.8)
- Se (numero_cluster= 1) allora numero_bips= 0 (3.0/2.8)
- Se (numero_cluster= 2) allora numero_bips= 5 (9.0/7.3)
- Se (numero_cluster= 3) allora numero_bips= 3 (12.0/9.5)
- Se (numero_cluster= 4) allora numero_bips= 3 (14.0/12.3)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora numero_bips= 1 (2.0/1.8)
- Se (numero_cluster= 1) allora numero_bips= 5 (4.0/3.1)

- Se (numero_cluster= 2) allora numero_bips= 0 (3.0/2.8)
- Se (numero_cluster= 3) allora numero_bips= 5 (9.0/7.3)
- Se (numero_cluster= 4) allora numero_bips= 3 (12.0/9.5)
- Se (numero_cluster= 5) allora numero_bips= 0 (10.0/9.1)

o)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora numero_BIPs= 1 (2.0/1.0)
- Se (numero_cluster= 1) allora numero_BIPs= 0 (3.0/2.0)
- Se ((numero_cluster= 2) & (avatar= cartoon) & (sesso= femmina)) allora numero_BIPs= 5 (3.0/1.0)
- Se ((numero_cluster= 2) & (avatar= cartoon) & (sesso= maschio)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 2) & (avatar= realistico) & (Variaz_GSR <= 1.028)) allora numero_BIPs= 4 (2.0)
- Se ((numero_cluster= 2) & (avatar= realistico) & (Variaz_GSR > 1.028)) allora numero_BIPs= 1 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 3 (2.0)
- Se ((numero_cluster= 3) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (avatar= cartoon)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (avatar= realistico) & (Variaz_GSR <= 3.14)) allora numero_BIPs= 3 (3.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (avatar= realistico) & (Variaz_GSR > 3.14)) allora numero_BIPs= 2 (3.0/2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= cartoon) & (Variaz_GSR <= 2.3)) allora numero_BIPs= 5 (2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= cartoon) & (2.3 < Variaz_GSR <= 9.702)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= cartoon) & (Variaz_GSR > 9.702)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_GSR <= 1.295)) allora numero_BIPs= 0 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (1.295 < Variaz_GSR <= 3.14)) allora numero_BIPs= 11 (2.0)

- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_GSR > 3.14)) allora numero_BIPs= 3
(2.0/1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora numero_BIPs= 1 (2.0/1.0)
- Se ((numero_cluster= 1) & (Variaz_GSR <= 3.14)) allora numero_BIPs= 5 (2.0)
- Se ((numero_cluster= 1) & (Variaz_GSR > 3.14)) allora numero_BIPs= 3 (2.0/1.0)
- Se (numero_cluster= 2) allora numero_BIPs= 0 (3.0/2.0)
- Se ((numero_cluster= 3) & (avatar= cartoon) & (sesso= femmina)) allora numero_BIPs= 5
(3.0/1.0)
- Se ((numero_cluster= 3) & (avatar= cartoon) & (sesso= maschio)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 3) & (avatar= realistico) & (Variaz_GSR <= 1.028)) allora numero_BIPs= 4
(2.0)
- Se ((numero_cluster= 3) & (avatar= realistico) & (Variaz_GSR > 1.028)) allora numero_BIPs= 1
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 3 (2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (avatar= cartoon)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (avatar= realistico) & (Variaz_GSR <= 3.14)) allora
numero_BIPs= 3 (3.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (avatar= realistico) & (Variaz_GSR > 3.14)) allora
numero_BIPs= 2 (3.0/2.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 2 (3.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (Variaz_GSR <= 1.295)) allora numero_BIPs= 0
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (Variaz_GSR > 1.295)) allora numero_BIPs= 11
(3.0/1.0)

P)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora numero_BIPs= 1 (2.0/1.0)
- Se (numero_cluster= 1) allora numero_BIPs= 0 (3.0/2.0)

- Se ((numero_cluster= 2) & (avatar= cartoon) & (sesso= femmina)) allora numero_BIPs= 5
(3.0/1.0)
- Se ((numero_cluster= 2) & (avatar= cartoon) & (sesso= maschio)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 2) & (avatar= realistico) & (Variaz_GSR <= 1.028)) allora numero_BIPs= 4
(2.0)
- Se ((numero_cluster= 2) & (avatar= realistico) & (Variaz_GSR > 1.028)) allora numero_BIPs= 1
(2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 3 (2.0)
- Se ((numero_cluster= 3) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0
(2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR <= -0.23) & (avatar= cartoon)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR <= -0.23) & (avatar= realistico)) allora numero_BIPs= 2
(4.0/2.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR > -0.23)) allora numero_BIPs= 6
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (Variaz_HR <= -1.14)) allora numero_BIPs= 0
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (Variaz_HR > -1.14) & (Variaz_GSR <=2.3)) allora numero_BIPs= 5
(2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (Variaz_HR > -1.14) & (2.3 < Variaz_GSR <= 9.702)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (Variaz_HR > -1.14) & (Variaz_GSR >= 9.702)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (VariazHr <= -0.44)) allora numero_BIPs= 3
(3.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (VariazHr > -0.44)) allora numero_BIPs= 0
(3.0/2.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora numero_BIPs= 1
(2.0/1.0)
- Se ((numero_cluster= 1) & (Variaz_GSR <= 3.14)) allora numero_BIPs= 5
(2.0)
- Se ((numero_cluster= 1) & (Variaz_GSR > 3.14)) allora numero_BIPs= 3
(2.0/1.0)
- Se (numero_cluster= 2) allora numero_BIPs= 0
(3.0/2.0)
- Se ((numero_cluster= 3) & (avatar= cartoon) & (sesso= femmina)) allora numero_BIPs= 5
(3.0/1.0)

- Se ((numero_cluster= 3) & (avatar= cartoon) & (sesso= maschio)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 3) & (avatar= realistico) & (Variaz_GSR <= 1.028)) allora numero_BIPs= 4
(2.0)
- Se ((numero_cluster= 3) & (avatar= realistico) & (Variaz_GSR > 1.028)) allora numero_BIPs= 1
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 3 (2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (avatar= cartoon) & (Variaz_HR <= -0.23)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (avatar= realistico) & (Variaz_HR <= -0.23)) allora numero_BIPs= 2
(4.0/2.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_HR > -0.23)) allora numero_BIPs= 6
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 2 (3.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= realistico)) allora numero_BIPs= 0
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (Variaz_GSR <= 1.295)) allora numero_BIPs= 0
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (Variaz_GSR > 1.295)) allora numero_BIPs= 11
(3.0/1.0)

Q)

➤ Caso: Clusters determinati con gerarchico e k= 5

- Se ((numero_cluster= 0) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 1) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 2) & (texture= alta) & (Variaz_GSR <= 0.844)) allora (4 <= numero_BIPs<=7)
(2.0)
- Se ((numero_cluster= 2) & (texture= alta) & (Variaz_GSR > 0.844)) allora (0 <=numero_BIPs<= 3)
(2.0/1.0)
- Se ((numero_cluster= 2) & (texture= bassa)) allora (0 <= numero_BIPs <= 3) (5.0/2.0)
- Se ((numero_cluster= 3) & (sesso= femmina)) allora (0 <= numero_BIPs <= 3) (2.0)
- Se ((numero_cluster= 3) & (sesso= maschio) & (texture= alta)) allora (0 <= numero_BIPs <=3)
(3.0/1.0)
- Se ((numero_cluster= 3) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR <= 3.14) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)

- Se ((numero_cluster= 3) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR <= 3.14) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 3) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR > 3.14)) allora (4<=numero_BIPs<= 7) (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= femmina) & (Variaz_GSR <= 1.91)) allora (0<=numero_BIPs<=3) (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= femmina) & (1.91 < Variaz_GSR <= 7.01)) allora (0 <= numero_BIPs <= 3) (2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= femmina) & (Variaz_GSR > 7.01)) allora (4<=numero_BIPs<= 7) (2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= maschio)) allora (0<= numero_BIPs<=3) (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_GSR <= 1.295)) allora (0 <= numero_BIPs <= 3) (2.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (1.295 < Variaz_GSR <= 3.14)) allora (8 <= numero_BIPs <= 11) (2.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_GSR > 3.14)) allora (0 <=numero_BIPs<= 3) (2.0/1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 1) & (Variaz_GSR <= 3.14)) allora (4 <= numero_BIPs<= 7) (2.0)
- Se ((numero_cluster= 1) & (Variaz_GSR > 3.14)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se (numero_cluster= 2) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 3) & (texture= alta) & (Variaz_GSR <= 0.844)) allora (4 <= numero_BIPs <= 7) (2.0)
- Se ((numero_cluster= 3) & (texture= alta) & (Variaz_GSR > 0.844)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa)) allora (0 <= numero_BIPs <= 3) (5.0/2.0)
- Se((numero_cluster= 4) & (sesso= femmina)) allora (0 <= numero_BIPs <= 3) (2.0)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= alta)) allora (0 <= numero_BIPs <= 3) (3.0/1.0)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR <= 3.14) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR <= 3.14) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR > 3.14)) allora (4 <= numero_BIPs <= 7) (2.0/1.0)

- Se ((numero_cluster= 5) & (texture= alta) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 3)
(3.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3)
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (Variaz_GSR <= 1.295)) allora (0 <= numero_BIPs <= 3)
(2.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (Variaz_GSR > 1.295)) allora (8 <= numero_BIPs <= 11)
(3.0/1.0)

R)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (numero_cluster= 0) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se (numero_cluster= 1) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 2) & (texture= alta) & (Variaz_GSR <= 0.844)) allora (4 <= numero_BIPs <= 7)
(2.0)
- Se ((numero_cluster= 2) & (texture= alta) & (Variaz_GSR > 0.844)) allora (0 <= numero_BIPs <= 3)
(2.0/1.0)
- Se ((numero_cluster= 2) & (texture= bassa) & (Variaz_HR <= -0.15)) allora (4 <= numero_BIPs <= 7)
(3.0/1.0)
- Se ((numero_cluster= 2) & (texture= bassa) & (Variaz_HR > -0.15)) allora (0 <= numero_BIPs <= 3)
(2.0)
- Se ((numero_cluster= 3) & (sesso= femmina)) allora (0 <= numero_BIPs <= 3) (2.0)
- Se ((numero_cluster= 3) & (sesso= maschio) & (texture= alta)) allora (0 <= numero_BIPs <= 3)
(3.0/1.0)
- Se ((numero_cluster= 3) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR <= 3.14) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 3) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR <= 3.14) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 3) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR > 3.14)) allora (4 <= numero_BIPs <= 7) (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (Variaz_HR <= -1.14)) allora (0 <= numero_BIPs <= 3)
(2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (-1.14 < Variaz_HR <= -0.21)) allora (4 <= numero_BIPs <= 7) (3.0)
- Se ((numero_cluster= 4) & (texture= alta) & (Variaz_HR > -0.21)) allora (0 <= numero_BIPs <= 3)
(3.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_GSR <= 1.295)) allora (0 <= numero_BIPs <= 3)
(2.0)

- Se ((numero_cluster= 4) & (texture= bassa) & (1.295 < Variaz_GSR <= 3.14)) allora (8 <= numero_BIPs <= 11) (2.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_GSR > 3.14)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 1) & (Variaz_GSR <= 3.14)) allora (4 <= numero_BIPs <= 7) (2.0)
- Se ((numero_cluster= 1) & (Variaz_GSR > 3.14)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se (numero_cluster= 2) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 3) & (texture= alta) & (Variaz_GSR <= 0.844)) allora (4 <= numero_BIPs <= 7) (2.0)
- Se ((numero_cluster= 3) & (texture= alta) & (Variaz_GSR > 0.844)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR <= -0.15)) allora (4 <= numero_BIPs <= 7) (3.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR > -0.15)) allora (0 <= numero_BIPs <= 3) (2.0)
- Se ((numero_cluster= 4) & (sesso= femmina)) allora (0 <= numero_BIPs <= 3) (2.0)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= alta)) allora (0 <= numero_BIPs <= 3) (3.0/1.0)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR <= 3.14) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR <= 3.14) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa) & (Variaz_GSR > 3.14)) allora (4 <= numero_BIPs <= 7) (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 3) (3.0)
- Se ((numero_cluster= 5) & (texture= alta) & (avatar= realistico)) allora (0 <= numero_BIPs <= 3) (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (Variaz_GSR <= 1.295)) allora (0 <= numero_BIPs <= 3) (2.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (Variaz_GSR > 1.295)) allora (8 <= numero_BIPs <= 11) (3.0/1.0)

S)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((numero_cluster= 0) allora (0<= numero_BIPs <= 1) (2.0/1.0)
- Se (numero_cluster= 1) allora (0 <= numero_BIPs <= 1) (3.0/1.0)
- Se ((numero_cluster= 2) & (texture= alta) & (Variaz_GSR <= 0.844)) allora (4 <= numero_BIPs <= 5) (2.0)
- Se ((numero_cluster= 2) & (texture= alta) & (Variaz_GSR > 0.844)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 2) & (texture= bassa)) allora (numero_BIPs= 2) (5.0/3.0)
- Se ((numero_cluster= 3) & (texture= alta) & (avatar= cartoon)) allora (numero_BIPs= 3) (2.0)
- Se ((numero_cluster= 3) & (texture= alta) & (avatar= realistico)) allora (0<= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (avatar= cartoon)) allora (numero_BIPs= 2) (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (avatar= realistico) & (Variaz_GSR <= 3.14)) allora (numero_BIPs= 3) (3.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (avatar= realistico) & (Variaz_GSR > 3.14)) allora (6 <= numero_BIPs <= 16) (3.0/1.0)
- Se ((numero_cluster= 4) & (Variaz_GSR <= 0.922)) allora (0 <= numero_BIPs <= 1) (3.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= femmina) & (0.922 <Variaz_GSR <= 3.14) allora (numero_BIPs= 2) (2.0/1.0)
- Se ((numero_cluster= 4) & (Variaz_GSR > 3.14) & (texture= alta) & (sesso= femmina)) allora (4 <= numero_BIPs<= 5) (3.0/1.0)
- Se ((numero_cluster= 4) & (Variaz_GSR > 0.922) & (texture= alta) & (sesso= maschio)) allora (numero_BIPs= 2) (2.0/1.0)
- Se ((numero_cluster= 4) & (0.922 < Variaz_GSR <= 3.14) & (texture= bassa)) allora (6 <= numero_BIPs <= 16) (2.0)
- Se ((numero_cluster= 4) & (Variaz_GSR > 3.14) & (texture= bassa)) allora (numero_BIPs= 3) (2.0/1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (numero_cluster= 0) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 1) & (Variaz_GSR <= 3.14)) allora (4 <= numero_BIPs <= 5) (2.0)
- Se ((numero_cluster= 1) & (Variaz_GSR > 3.14)) allora (numero_BIPs= 3) (2.0/1.0)
- Se (numero_cluster= 2) allora (0 <= numero_BIPs <= 1) (3.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (Variaz_GSR <= 0.844)) allora (4 <= numero_BIPs <= 5) (2.0)

- Se ((numero_cluster= 3) & (texture= alta) & (Variaz_GSR > 0.844)) allora (0 <= numero_BIPs <= 1)
(2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa)) allora numero_cluster= 2 (5.0/3.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 3 (2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1)
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (avatar= cartoon)) allora numero_BIPs= 2
(2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (avatar= realistico) & (Variaz_GSR <= 3.14)) allora
numero_BIPs= 3 (3.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (avatar= realistico) & (Variaz_GSR > 3.14)) allora (6 <= numero_BIPs <= 16)
(3.0/1.0)
- Se ((numero_cluster= 5) & (Variaz_GSR <= 1.295)) allora (0 <= numero_BIPs <= 1) (3.0/1.0)
- Se ((numero_cluster= 5) & (1.295 < Variaz_GSR <= 3.073)) allora (6 <= numero_BIPs <= 16)
(2.0)
- Se ((numero_cluster= 5) & (3.073 <= Variaz_GSR <= 7.008)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 5) & (Variaz_GSR > 7.01)) allora (4 <= numero_BIPs <= 5) (3.0/1.0)

T)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se ((numero_cluster= 0) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se (numero_cluster= 1) allora (0 <= numero_BIPs <= 1) (3.0/1.0)
- Se ((numero_cluster= 2) & (texture= alta) & (Variaz_GSR <= 0.844)) allora (4 <= numero_BIPs <= 5)
(2.0)
- Se ((numero_cluster= 2) & (texture= alta) & (Variaz_GSR > 0.844)) allora (0 <= numero_BIPs <= 1)
(2.0/1.0)
- Se ((numero_cluster= 2) & (texture= bassa) & (Variaz_HR <= -0.15)) allora (4 <= numero_BIPs <= 5)
(3.0/1.0)
- Se ((numero_cluster= 2) & (texture= bassa) & (Variaz_HR > -0.15)) allora (numero_BIPs= 2)
(2.0)
- Se ((numero_cluster= 3) & (texture= alta) & (avatar= cartoon)) allora (numero_BIPs= 3)
(2.0)
- Se ((numero_cluster= 3) & (texture= alta) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1)
(2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR <= -1.19)) allora numero_BIPs= 2
(2.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR > -1.19) & (Variaz_GSR <= 0.399)) allora
numero_BIPs= 2 (2.0/1.0)

- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR > -1.19) & (0.399 < Variaz_GSR <= 3.14)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR > -1.19) & (Variaz_GSR > 3.14)) allora (6 <= numero_BIPs <= 16) (2.0)
- Se ((numero_cluster= 4) & (Variaz_GSR <= 0.922)) allora (0 <= numero_BIPs <= 1) (3.0/1.0)
- Se ((numero_cluster= 4) & (texture= alta) & (Variaz_HR <= -0.21) & (0.922 < Variaz_GSR <= 3.14)) allora (4 <= numero_BIPs <= 5) (2.0)
- Se ((numero_cluster= 4) & (Variaz_GSR > 3.14) & (texture= alta) & (Variaz_HR <= -0.21)) allora (numero_BIPs= 3) (2.0/1.0)
- Se ((numero_cluster= 4) & (Variaz_GSR > 0.922) & (texture= alta) & (Variaz_HR > -0.21)) allora numero_BIPs= 2 (3.0/1.0)
- Se ((numero_cluster= 4) & (0.922 < Variaz_GSR <= 3.14) & (texture= bassa)) allora (6 <= numero_BIPs <= 16) (2.0)
- Se ((numero_cluster= 4) & (Variaz_GSR > 3.14) & (texture= bassa)) allora numero_BIPs= 3 (2.0/1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se ((numero_cluster= 0) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 1) & (Variaz_GSR <= 3.14)) allora (4 <= numero_BIPs <= 5) (2.0)
- Se ((numero_cluster= 1) & (Variaz_GSR > 3.14)) allora numero_BIPs= 3 (2.0/1.0)
- Se (numero_cluster= 2) allora (0 <= numero_BIPs <= 1) (3.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (Variaz_GSR <= 0.844)) allora (4 <= numero_BIPs <= 5) (2.0)
- Se ((numero_cluster= 3) & (texture= alta) & (Variaz_GSR > 0.844)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR <= -0.15)) allora (4 <= numero_BIPs <= 5) (3.0/1.0)
- Se ((numero_cluster= 3) & (texture= bassa) & (Variaz_HR > -0.15)) allora numero_BIPs= 2 (2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= cartoon)) allora numero_BIPs= 3 (2.0)
- Se ((numero_cluster= 4) & (texture= alta) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_HR <= -1.19)) allora numero_BIPs= 2 (2.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_HR > -1.19) & (Variaz_GSR <= 0.40)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_HR > -1.19) & (0.40 < Variaz_GSR <= 3.14)) allora numero_BIPs= 3 (2.0/1.0)

- Se ((numero_cluster= 4) & (texture= bassa) & (Variaz_HR > -1.19) & (Variaz_GSR > 3.14)) allora (6 <= numero_BIPs <= 16) (2.0)
- Se ((numero_cluster= 5) & (Variaz_HR <= -1.19)) allora numero_BIPs= 3 (2.0)
- Se ((numero_cluster= 5) & (Variaz_HR > -1.19) & (Variaz_GSR <= 1.295)) allora (0 <= numero_BIPs <= 1) (2.0)
- Se ((numero_cluster= 5) & (Variaz_HR > -1.19) & (1.295 <= Variaz_GSR <= 3.073)) allora (6 <= numero_BIPs <= 16) (2.0)
- Se ((numero_cluster= 5) & (Variaz_HR > -1.19) & (Variaz_GSR > 3.073)) allora numero_BIPs= 2 (4.0/2.0)

U)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (Age= 18) allora numero_BIPs= 0 (3.0/2.0)
- Se ((Age= 19) & (Variaz_GSR <= 2.13)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((Age= 19) & (Variaz_GSR > 2.13)) allora numero_BIPs= 2 (2.0/1.0)
- Se (Age= 20) allora numero_BIPs= 0 (3.0/2.0)
- Se (Age= 21) allora numero_BIPs= 0 (3.0/2.0)
- Se (Age= 22) allora numero_BIPs= 0 (1.0)
- Se (Age= 23) allora numero_BIPs= 1 (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR <= 0.616)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR > 0.616)) allora numero_BIPs= 0 (2.0/1.0)
- Se (Age= 25) allora numero_BIPs= 1 (3.0/2.0)
- Se (Age= 26) allora numero_BIPs= 2 (2.0/1.0)
- Se (Age= 27) allora numero_BIPs= 5 (2.0)
- Se (Age= 28) allora numero_BIPs= 3 (3.0/2.0)
- Se (Age= 29) allora numero_BIPs= 5 (1.0)
- Se (Age= 30) allora numero_BIPs= 3 (2.0)
- Se (Age= 32) allora numero_BIPs= 2 (1.0)
- Se (Age= 34) allora numero_BIPs= 3 (2.0/1.0)
- Se (Age= 35) allora numero_BIPs= 2 (1.0)
- Se (Age= 40) allora numero_BIPs= 4 (1.0)
- Se (Age= 41) allora numero_BIPs= 4 (1.0)
- Se (Age= 43) allora numero_BIPs= 11 (1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (Age= 18) allora numero_BIPs= 0 (3.0/2.0)
- Se ((Age= 19) & (Variaz_GSR <= 2.13)) allora numero_BIPs= 3 (2.0/1.0)

- Se ((Age= 19) & (Variaz_GSR > 2.13)) allora numero_BIPs= 2 (2.0/1.0)
- Se (Age= 20) allora numero_BIPs= 0 (3.0/2.0)
- Se (Age= 21) allora numero_BIPs= 0 (3.0/2.0)
- Se (Age= 22) allora numero_BIPs= 0 (1.0)
- Se (Age= 23) allora numero_BIPs= 1 (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR <= 0.616)) allora numero_BIPs= 3 (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR > 0.616)) allora numero_BIPs= 0 (2.0/1.0)
- Se (Age= 25) allora numero_BIPs= 1 (3.0/2.0)
- Se (Age= 26) allora numero_BIPs= 2 (2.0/1.0)
- Se (Age= 27) allora numero_BIPs= 5 (2.0)
- Se (Age= 28) allora numero_BIPs= 3 (3.0/2.0)
- Se (Age= 29) allora numero_BIPs= 5 (1.0)
- Se (Age= 30) allora numero_BIPs= 3 (2.0)
- Se (Age= 32) allora numero_BIPs= 2 (1.0)
- Se (Age= 34) allora numero_BIPs= 3 (2.0/1.0)
- Se (Age= 35) allora numero_BIPs= 2 (1.0)
- Se (Age= 40) allora numero_BIPs= 4 (1.0)
- Se (Age= 41) allora numero_BIPs= 4 (1.0)
- Se (Age= 43) allora numero_BIPs= 11 (1.0)

V)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (Age= 18) allora (0 <=numero_BIPs<=3) (3.0)
- Se ((Age= 19) & (Variaz_HR <= -0.3)) allora(0 <= numero_BIPs<= 3) (2.0)
- Se ((Age= 19) & (Variaz_HR > -0.3)) allora (4<=numero_BIPs<= 7) (2.0)
- Se (Age= 20) allora (0 <=numero_BIPs<= 3) (3.0/1.0)
- Se (Age= 21) allora (0 <=numero_BIPs<= 3) (3.0/1.0)
- Se (Age= 22) allora (0 <=numero_BIPs<= 3) (1.0)
- Se (Age= 23) allora (0<=numero_BIPs<= 3) (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR <= 0.616)) allora (0<=numero_BIPs<= 3) (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR > 0.616)) allora (0<=numero_BIPs<= 3) (2.0)
- Se (Age= 25) allora (0<=numero_BIPs<= 3) (3.0/1.0)
- Se (Age= 26) allora (0<=numero_BIPs<= 3) (2.0/1.0)
- Se (Age= 27) allora (4<=numero_BIPs<= 7) (2.0)
- Se (Age= 28) allora (4<=numero_BIPs<= 7) (3.0/1.0)
- Se (Age= 29) allora (4<=numero_BIPs<= 7) (1.0)
- Se (Age= 30) allora (0<=numero_BIPs<= 3) (2.0)
- Se (Age= 32) allora (0<=numero_BIPs<= 3) (1.0)

- Se (Age= 34) allora (0<=numero_BIPs<= 3) (2.0/1.0)
- Se (Age= 35) allora (0<=numero_BIPs<= 3) (1.0)
- Se (Age= 40) allora (4<=numero_BIPs<= 7) (1.0)
- Se (Age= 41) allora (4<=numero_BIPs<= 7) (1.0)
- Se (Age= 43) allora (8<=numero_BIPs<= 11) (1.0)

➤ *Caso: Clusters determinati con gerarchico e k= 6*

- Se (Age= 18) allora (0 <=numero_BIPs<=3) (3.0)
- Se ((Age= 19) & (Variatz_HR <= -0.3)) allora(0 <= numero_BIPs<= 3) (2.0)
- Se ((Age= 19) & (Variatz_HR > -0.3)) allora (4<=numero_BIPs<= 7) (2.0)
- Se (Age= 20) allora (0 <=numero_BIPs<= 3) (3.0/1.0)
- Se (Age= 21) allora (0 <=numero_BIPs<= 3) (3.0/1.0)
- Se (Age= 22) allora (0 <=numero_BIPs<= 3) (1.0)
- Se (Age= 23) allora (0<=numero_BIPs<= 3) (2.0/1.0)
- Se ((Age= 24) & (Variatz_GSR <= 0.616)) allora (0<=numero_BIPs<= 3) (2.0/1.0)
- Se ((Age= 24) & (Variatz_GSR > 0.616)) allora (0<=numero_BIPs<= 3) (2.0)
- Se (Age= 25) allora (4<=numero_BIPs<= 7) (3.0/1.0)
- Se (Age= 26) allora (0<=numero_BIPs<= 3) (2.0/1.0)
- Se (Age= 27) allora (4<=numero_BIPs<= 7) (2.0)
- Se (Age= 28) allora (4<=numero_BIPs<= 7) (3.0/1.0)
- Se (Age= 29) allora (4<=numero_BIPs<= 7) (1.0)
- Se (Age= 30) allora (0<=numero_BIPs<= 3) (2.0)
- Se (Age= 32) allora (0<=numero_BIPs<= 3) (1.0)
- Se (Age= 34) allora (0<=numero_BIPs<= 3) (2.0/1.0)
- Se (Age= 35) allora (0<=numero_BIPs<= 3) (1.0)
- Se (Age= 40) allora (4<=numero_BIPs<= 7) (1.0)
- Se (Age= 41) allora (4<=numero_BIPs<= 7) (1.0)
- Se (Age= 43) allora (8<=numero_BIPs<= 11) (1.0)

Z)

➤ *Caso: Clusters determinati con gerarchico e k= 5*

- Se (Age= 18) allora (0 <=numero_BIPs<=1) (3.0/2.0)
- Se ((Age= 19) & (Variatz_HR <= -0.3)) allora(numero_BIPs= 2) (2.0/1.0)
- Se ((Age= 19) & (Variatz_HR > -0.3)) allora (4<=numero_BIPs<= 5) (2.0)
- Se (Age= 20) allora (0 <=numero_BIPs<= 1) (3.0/2.0)
- Se (Age= 21) allora (0 <=numero_BIPs<= 1) (3.0/2.0)
- Se (Age= 22) allora (0 <=numero_BIPs<= 1) (1.0)
- Se (Age= 23) allora (0<=numero_BIPs<= 1) (2.0/1.0)

- Se ((Age= 24) & (Variaz_GSR <= 0.616)) allora (numero_BIPs= 3) (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR > 0.616)) allora (0<=numero_BIPs<= 1) (2.0)
- Se (Age= 25) allora (0<=numero_BIPs<= 1) (3.0/2.0)
- Se (Age= 26) allora (numero_BIPs= 2) (2.0/1.0)
- Se (Age= 27) allora (4<=numero_BIPs<= 5) (2.0)
- Se (Age= 28) allora (numero_BIPs= 3) (3.0/2.0)
- Se (Age= 29) allora (4<=numero_BIPs<= 5) (1.0)
- Se (Age= 30) allora (numero_BIPs= 3) (2.0)
- Se (Age= 32) allora (numero_BIPs= 2) (1.0)
- Se (Age= 34) allora (numero_BIPs= 3) (2.0/1.0)
- Se (Age= 35) allora (numero_BIPs= 2) (1.0)
- Se (Age= 40) allora (4<=numero_BIPs<= 5) (1.0)
- Se (Age= 41) allora (4<=numero_BIPs<= 5) (1.0)
- Se (Age= 43) allora (6=numero_BIPs<= 16) (1.0)


➤ *Caso: Clusters determinati con gerarchico e k= 6*


- Se (Age= 18) allora (0 <=numero_BIPs<=1) (3.0/2.0)
- Se ((Age= 19) & (Variaz_HR <= -0.3)) allora(numero_BIPs= 2) (2.0/1.0)
- Se ((Age= 19) & (Variaz_HR > -0.3)) allora (4<=numero_BIPs<= 5) (2.0)
- Se (Age= 20) allora (0 <=numero_BIPs<= 1) (3.0/2.0)
- Se (Age= 21) allora (0 <=numero_BIPs<= 1) (3.0/2.0)
- Se (Age= 22) allora (0 <=numero_BIPs<= 1) (1.0)
- Se (Age= 23) allora (0<=numero_BIPs<= 1) (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR <= 0.616)) allora (numero_BIPs= 3) (2.0/1.0)
- Se ((Age= 24) & (Variaz_GSR > 0.616)) allora (0<=numero_BIPs<= 1) (2.0)
- Se (Age= 25) allora (0<=numero_BIPs<= 1) (3.0/2.0)
- Se (Age= 26) allora (numero_BIPs= 2) (2.0/1.0)
- Se (Age= 27) allora (4<=numero_BIPs<= 5) (2.0)
- Se (Age= 28) allora (numero_BIPs= 3) (3.0/2.0)
- Se (Age= 29) allora (4<=numero_BIPs<= 5) (1.0)
- Se (Age= 30) allora (numero_BIPs= 3) (2.0)
- Se (Age= 32) allora (numero_BIPs= 2) (1.0)
- Se (Age= 34) allora (numero_BIPs= 3) (2.0/1.0)
- Se (Age= 35) allora (numero_BIPs= 2) (1.0)
- Se (Age= 40) allora (4<=numero_BIPs<= 5) (1.0)
- Se (Age= 41) allora (4<=numero_BIPs<= 5) (1.0)
- Se (Age= 43) allora (6=numero_BIPs<= 16) (1.0)

		A	B	C	D	E	F	G	H	I	L	M	N	O	P	Q	R	S	T	U	V	Z
Clustering																						
Bips	Valori																					
	Int.regolari																					
	Int.irregolari																					
Variazione % GSR																						
Variazione Heart Rate																						
Texture																						
Avatar																						
Sesso																						
Altri Dati(*)																						

Tabella C-1 Attributi utilizzati in fase di classificazioni

Legenda

 target

 attributo

(*) AltriDati: Occupy/ Medicine/ Language/ Alcohol/
Literate/ Programming/ VReality/ Games/ Gametimes

Appendice D

L'utilizzo dell'approccio definito nel paragrafo 6.9 ci ha consentito di realizzare una fase di clustering e una di classificazione. La suddivisione delle traiettorie in clusters è stata fatta sulla base della similarità espressa in termini di coordinate spaziali e di parametri fisiologici. La rappresentazioni di questi attributi in sistemi di riferimento diversi ha reso necessario la normalizzazione dei valori. Come già spiegato in maniera più dettagliata nel paragrafo 6.9, per la normalizzazione sono stati impiegati due approcci diversi: lo z-score normalization e il min-max normalization.

Anche in questo caso è stata riscontrata una certa stabilità nei k clusters con $k=5$ e $k=6$, facilitando il confronto con i risultati ottenuti nelle altre fasi del nostro studio

I due approcci hanno portato ad individuare due suddivisioni diverse in clusters. La fase successiva di classificazione è stata realizzata considerando entrambe. In tabella D-1 è riportato per ciascun individuo il cluster di appartenenza a seconda dell'approccio utilizzato.

Per realizzare il clustering è stato utilizzato l'algoritmo gerarchico con i dati in input normalizzati rispettivamente con lo z-score normalization e il min-max normalization.

	z-score		Min- max	
Idc	5	6	5	6
11111	4	5	1	1
11121	4	5	1	1
11211	1	1	1	1
11221	4	5	1	1
12111	4	5	1	1
12121	4	5	3	5
12211	4	4	3	5
12221	4	5	3	5
21111	4	5	1	1
21121	4	4	1	1
21211	4	5	3	5
21221	4	5	1	1
22111	4	4	0	0
22121	2	2	4	2
22211	4	4	3	5
22221	1	1	1	1
31111	3	3	4	3
31121	4	4	0	0
31211	3	3	2	4
31221	3	3	4	3
32111	3	3	4	3
32121	3	3	4	3
32211	4	5	3	5
32221	2	2	4	2
41111	3	3	2	4
41121	4	5	1	1
41211	3	3	4	3
41221	4	4	0	0
42111	4	5	0	0
42121	4	5	3	5
42211	2	2	4	2
42221	2	2	4	2
51111	4	4	4	2
51121	4	4	0	0
51211	2	2	3	5
51221	3	3	2	4
52111	4	4	0	0
52121	3	3	2	4
52211	4	4	0	0
52221	0	0	2	4

Tabella D-1:Suddivisione delle traiettorie in clusters

A)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- (numero_cluster= 1) (40.0/32.2)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- (numero_cluster= 1) (40.0/32.2)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((numero_BIPs= 0) allora numero_cluster= 2 (5.0/4.1)
- Se ((numero_BIPs= 1) allora numero_cluster= 5 (3.0/2.1)
- Se ((numero_BIPs= 2) allora numero_cluster= 3 (8.0/5.4)
- Se ((numero_BIPs= 3) allora numero_cluster= 4 (7.0/3.4)
- Se ((numero_BIPs= 4) allora numero_cluster= 1 (4.0/3.8)
- Se ((numero_BIPs= 5) allora numero_cluster= 4 (6.0/5.1)
- Se ((numero_BIPs= 6) allora numero_cluster= 5 (1.0/0.8)
- Se ((numero_BIPs= 8) allora numero_cluster= 3 (2.0/1.8)
- Se ((numero_BIPs= 10) allora numero_cluster= 3 (1.0/0.8)
- Se ((numero_BIPs= 11) allora numero_cluster= 5 (2.0/1.0)
- Se ((numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

B)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se ((sesso= femmina) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 3 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 0 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= realistico)) allora numero_cluster= 4 (10.0/7.5)
- Se (sesso= maschio) allora numero_cluster= 1 (20.0/13.9)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se ((sesso= femmina) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 5 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 0 (5.0/3.2)

- Se ((sesso= femmina) & (avatar= realistico) & (texture= alta)) allora numero_cluster= 2
(5.0/4.1)
- Se ((sesso= femmina) & (avatar= realistico) & (texture= bassa)) allora numero_cluster= 5
(5.0/4.1)
- Se (sesso= maschio) allora numero_cluster= 1
(20.0/13.9)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((texture= alta) & (sesso= femmina) & (avatar= cartoon)) allora numero_cluster= 4 (5.0/3.2)
- Se ((texture= alta) & (sesso= femmina) & (avatar= realistico)) allora numero_cluster= 2
(5.0/4.1)
- Se ((texture= alta) & (sesso= maschio)) allora numero_cluster= 3 (10.0/7.5)
- Se (texture= bassa) allora numero_cluster= 5 (20.0/13.9)

C)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Media_HR <= 76.05) allora numero_cluster= 2 (7.0/4.4)
- Se ((76.05 < Media_HR <= 93.13) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 1
(6.0/3.3)
- Se ((Media_HR > 93.13) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 4
(4.0/3.1)
- Se ((76.05 < Media_HR <= 87.69) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 4
(5.0/3.2)
- Se ((Media_HR > 87.69) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 3
(3.0/2.1)
- Se ((76.05 < Media_HR <= 84.81) & (sesso= maschio)) allora numero_cluster= 0 (3.0/2.1)
- Se ((84.81 < Media_HR <= 89.95) & (sesso= maschio) & (avatar= cartoon)) allora numero_cluster= 4
(3.0/2.1)
- Se ((Media_HR > 89.95) & (sesso= maschio) & (avatar= cartoon)) allora numero_cluster= 1
(3.0/2.1)
- Se ((Media_HR > 84.81) & (sesso= maschio) & (avatar= realistico)) allora numero_cluster= 1
(6.0/2.3)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Media_HR <= 76.05) allora numero_cluster= 4 (7.0/4.4)

- Se ((76.05 < Media_HR <= 93.13) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 1
(6.0/3.3)
- Se ((Media_HR > 93.13) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 2
(4.0/3.1)
- Se ((76.05 < Media_HR <= 87.69) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 2
(5.0/4.1)
- Se ((Media_HR > 87.69) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 5
(3.0/2.1)
- Se ((76.05 < Media_HR <= 84.81) & (sesso= maschio)) allora numero_cluster= 0
(3.0/2.1)
- Se ((Media_HR > 84.81) & (sesso= maschio)) allora numero_cluster= 1
(12.0/6.7)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((Media_HR <= 89.62) & (avatar= cartoon) & (sesso= femmina)) allora numero_cluster= 4
(6.0/3.3)
- Se ((Media_HR <= 87.69) & (avatar= cartoon) & (sesso= maschio)) allora numero_cluster= 3
(5.0/3.2)
- Se ((87.69 < Media_HR <= 89.62) & (avatar= cartoon) & (sesso= maschio)) allora numero_cluster= 2
(2.0/1.8)
- Se ((Media_HR <= 89.62) & (avatar= realistico) & (sesso= femmina)) allora numero_cluster= 3
(4.0/3.1)
- Se ((Media_HR <= 89.62) & (avatar= realistico) & (sesso= maschio)) allora numero_cluster= 4
(5.0/2.3)
- Se (89.62 < Media_HR <= 108.38) allora numero_cluster= 5
(15.0/6.8)
- Se (Media_HR > 108.38) allora numero_cluster= 3
(3.0/2.1)

D)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (sesso= femmina) allora numero_cluster= 3 (20.0/15.8)
- Se (sesso= maschio) allora numero_cluster= 0 (20.0/13.9)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (sesso= femmina) allora numero_cluster= 5 (20.0/15.8)
- Se (sesso= maschio) allora numero_cluster= 0 (20.0/13.9)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((numero_BIPs= 0) & (texture= alta)) allora numero_cluster= 2 (3.0/2.1)
- Se ((numero_BIPs= 0) & (texture= bassa)) allora numero_cluster= 5 (2.0/1.0)
- Se (numero_BIPs= 1) allora numero_cluster= 5 (3.0/2.1)
- Se (numero_BIPs= 2) allora numero_cluster= 3 (8.0/5.4)
- Se (numero_BIPs= 3) allora numero_cluster= 4 (7.0/3.4)
- Se ((numero_BIPs= 4) & (avatar= cartoon)) allora numero_cluster= 2 (2.0/1.8)
- Se ((numero_BIPs= 4) & (avatar= realistico)) allora numero_cluster= 1 (2.0/1.8)
- Se ((numero_BIPs= 5) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 5 (2.0/1.0)
- Se ((numero_BIPs= 5) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 4 (2.0/1.0)
- Se ((numero_BIPs= 5) & (sesso= maschio)) allora numero_cluster= 2 (2.0/1.8)
- Se (numero_BIPs= 6) allora numero_cluster= 5 (1.0/0.8)
- Se (numero_BIPs= 8) allora numero_cluster= 3 (2.0/1.8)
- Se (numero_BIPs= 10) allora numero_cluster= 3 (1.0/0.8)
- Se (numero_BIPs= 11) allora numero_cluster= 5 (2.0/1.0)
- Se (numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

E)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Media_HR < 75.55) allora numero_cluster= 2 (6.0/3.3)
- Se ((Media_HR > 75.55) & (sesso= femmina)) allora numero_cluster= 3 (19.0/14.8)
- Se ((Media_HR > 75.55) & (sesso= maschio)) allora numero_cluster= 0 (15.0/9.8)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Media_HR < 75.55) allora numero_cluster= 4 (6.0/3.3)
- Se ((Media_HR > 75.55) & (sesso= femmina)) allora numero_cluster= 5 (19.0/14.8)
- Se ((Media_HR > 75.55) & (sesso= maschio)) allora numero_cluster= 0 (15.0/9.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- Se (Media_HR <= 75.55) allora numero_cluster= 3 (6.0/3.3)
- Se (Media_HR > 75.55) allora numero_cluster= 4 (34.0/15.5)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se (Media_HR <= 75.55) allora numero_cluster= 3 (6.0/3.3)
- Se (75.55 < Media_HR <= 87.69) allora numero_cluster= 4 (14.0/8.8)
- Se (Media_HR > 87.69) allora numero_cluster= 5 (20.0/12.0)

F)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se ((sesso= femmina) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 3 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 0 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= realistico) & (0 <=numero_BIPs <= 3) allora numero_cluster= 4 (6.0/3.3)
- Se ((sesso= femmina) & (avatar= realistico) & (4 <=numero_BIPs <= 7) allora numero_cluster= 1 (2.0/1.8)
- Se ((sesso= femmina) & (avatar= realistico) & (8 <=numero_BIPs <= 11) allora numero_cluster= 3 (1.0/0.8)
- Se ((sesso= femmina) & (avatar= realistico) & (numero_BIPs= 16) allora numero_cluster= 2 (1.0/0.8)
- Se (sesso= maschio) allora numero_cluster= 1 (20.0/13.9)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se ((sesso= femmina) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 5 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 0 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= realistico) & (0 <=numero_BIPs <= 3) allora numero_cluster= 2 (6.0/4.3)
- Se ((sesso= femmina) & (avatar= realistico) & (4 <=numero_BIPs <= 7) allora numero_cluster= 1 (2.0/1.8)
- Se ((sesso= femmina) & (avatar= realistico) & (8 <=numero_BIPs <= 11) allora numero_cluster= 5 (1.0/0.8)
- Se ((sesso= femmina) & (avatar= realistico) & (numero_BIPs= 16) allora numero_cluster= 4 (1.0/0.8)
- Se (sesso= maschio) allora numero_cluster= 1 (20.0/13.9)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((0 <= numero_BIPs <= 3) & (avatar= cartoon) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 4 (3.0/1.1)
- Se ((0 <= numero_BIPs <= 3) & (avatar= cartoon) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 5 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 3) & (avatar= cartoon) & (sesso= maschio)) allora numero_cluster= 3 (6.0/5.1)
- Se ((0 <= numero_BIPs <= 3) & (avatar= realistico) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 2 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 3) & (avatar= realistico) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 3 (4.0/3.1)
- Se ((0 <= numero_BIPs <= 3) & (avatar= realistico) & (sesso= maschio)) allora numero_cluster= 4 (6.0/3.3)
- Se ((4 <= numero_BIPs <= 7) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 2 (4.0/3.1)
- Se ((4 <= numero_BIPs <= 7) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 3 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 7) & (avatar= realistico)) allora numero_cluster= 5 (4.0/2.2)
- Se (8 <= numero_BIPs <= 11) allora numero_cluster= 5 (5.0/3.2)
- Se (numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

G)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Media_HR <= 76.05) allora numero_cluster= 2 (7.0/4.4)
- Se ((76.05 < Media_HR <= 84.81) & (texture= alta)) allora numero_cluster= 3 (3.0/1.1)
- Se ((76.05 < Media_HR <= 84.81) & (texture= bassa) & (sesso= femmina)) allora numero_cluster= 4 (5.0/3.2)
- Se ((76.05 < Media_HR <= 84.81) & (texture= bassa) & (sesso= maschio)) allora numero_cluster= 0 (2.0/1.0)
- Se ((Media_HR > 84.81) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 4 (8.0/6.3)
- Se ((Media_HR > 84.81) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 1 (3.0/2.1)
- Se ((Media_HR > 84.81) & (sesso= maschio)) allora numero_cluster= 1 (12.0/6.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Media_HR <= 76.05) allora numero_cluster= 4 (7.0/4.4)
- Se ((76.05 < Media_HR <= 84.81) & (texture= alta)) allora numero_cluster= 5 (3.0/1.1)

- Se ((76.05 < Media_HR <= 84.81) & (texture= bassa)) allora numero_cluster= 0 (7.0/4.4)
- Se ((84.81 < Media_HR <= 100.32) & (sesso= femmina)) allora numero_cluster= 1 (8.0/5.4)
- Se ((100.32 < Media_HR <= 109.9) & (sesso= femmina)) allora numero_cluster= 2 (2.0/1.0)
- Se ((84.81 < Media_HR <= 109.9) & (sesso= maschio)) allora numero_cluster= 1 (11.0/5.6)
- Se (Media_HR > 109.9) allora numero_cluster= 0 (2.0/1.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((0 <= numero_BIPs <= 3) & (avatar= cartoon) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 4 (3.0/1.1)
- Se ((0 <= numero_BIPs <= 3) & (avatar= cartoon) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 5 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 3) & (avatar= cartoon) & (sesso= maschio)) allora numero_cluster= 3 (6.0/5.1)
- Se ((0 <= numero_BIPs <= 3) & (avatar= realistico) & (sesso= femmina) & (texture= alta)) allora numero_cluster= 2 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 3) & (avatar= realistico) & (sesso= femmina) & (texture= bassa)) allora numero_cluster= 3 (4.0/3.1)
- Se ((0 <= numero_BIPs <= 3) & (avatar= realistico) & (sesso= maschio)) allora numero_cluster= 4 (6.0/3.3)
- Se ((4 <= numero_BIPs <= 7) & (Media_HR <= 87.69) & (texture= alta)) allora numero_cluster= 1 (2.0/1.8)
- Se ((4 <= numero_BIPs <= 7) & (Media_HR <= 87.69) & (texture= bassa)) allora numero_cluster= 4 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 7) & (87.69 < Media_HR <= 89.63)) allora numero_cluster= 2 (2.0/1.0)
- Se ((4 <= numero_BIPs <= 7) & (Media_HR > 89.63)) allora numero_cluster= 5 (4.0/1.2)
- Se (8 <= numero_BIPs <= 11) allora numero_cluster= 5 (5.0/3.2)
- Se (numero_BIPs= 16) allora numero_cluster= 0 (1.0/0.8)

H)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se ((sesso= femmina) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 3 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= cartoon) & (texture= bassa)) allora numero_cluster= 0 (5.0/3.2)
- Se ((sesso= femmina) & (avatar= realistico)) allora numero_cluster= 4 (10.0/7.5)

- Se (sesso= maschio) allora numero_cluster= 1 (20.0/13.9)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se ((texture= alta) & (avatar= cartoon)) allora numero_cluster= 5 (10.0/6.5)
- Se ((texture= alta) & (avatar= realistico)) allora numero_cluster= 1 (10.0/8.3)
- Se (texture= bassa) allora numero_cluster= 0 (20.0/15.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((0 <= numero_BIPs <= 1) & (avatar= cartoon)) allora numero_cluster= 5 (3.0/1.1)
- Se ((0 <= numero_BIPs <= 1) & (avatar= realistico) & (sesso= femmina)) allora numero_cluster= 2 (2.0/1.0)
- Se ((0 <= numero_BIPs <= 1) & (avatar= realistico) & (sesso= maschio)) allora numero_cluster= 3 (3.0/2.8)
- Se (numero_BIPs= 2) allora numero_cluster= 3 (8.0/5.4)
- Se (numero_BIPs= 3) allora numero_cluster= 4 (7.0/3.4)
- Se ((4 <= numero_BIPs <= 5) & (avatar= cartoon) & (texture= alta)) allora numero_cluster= 2 (4.0/3.1)
- Se ((4 <= numero_BIPs <= 5) & (avatar= realistico) & (texture= alta)) allora numero_cluster= 5 (3.0/2.1)
- Se ((4 <= numero_BIPs <= 5) & (texture= bassa)) allora numero_cluster= 4 (3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (texture= alta)) allora numero_cluster= 0 (2.0/1.8)
- Se ((6 <= numero_BIPs <= 16) & (texture= bassa)) allora numero_cluster= 5 (5.0/2.3)

I)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Media_HR <= 76.05) allora numero_cluster= 2 (7.0/4.4)
- Se ((76.05 < Media_HR <= 89.63) & (sesso= femmina)) allora numero_cluster= 4 (9.0/6.4)
- Se ((Media_HR > 89.63) & (sesso= femmina)) allora numero_cluster= 3 (9.0/6.4)
- Se ((Media_HR > 76.05) & (sesso= maschio)) allora numero_cluster= 1 (15.0/9.8)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Media_HR <= 76.05) allora numero_cluster= 4 (7.0/4.4)

- Se ((Media_HR > 76.05) & (texture= alta) & (avatar= cartoon)) allora numero_cluster= 5
(9.0/5.5)
- Se ((Media_HR > 76.05) & (texture= alta) & (avatar= realistico)) allora numero_cluster= 1
(8.0/6.3)
- Se ((Media_HR > 76.05) & (texture= bassa)) allora numero_cluster= 0
(16.0/12.6)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((0 <= numero_BIPs <= 1) & (Media_HR <= 77.08)) allora numero_cluster= 3 (2.0/1.8)
- Se ((0 <= numero_BIPs <= 1) & (Media_HR > 77.08) & (avatar= cartoon)) allora numero_cluster= 5
(3.0/1.1)
- Se ((0 <= numero_BIPs <= 1) & (Media_HR > 77.08) & (avatar= realistico)) allora numero_cluster= 2
(3.0/2.1)
- Se ((numero_BIPs= 2) & (Media_HR <= 75.55)) allora numero_cluster= 3 (2.0/1.0)
- Se ((numero_BIPs= 2) & (75.55 < Media_HR <= 87.69)) allora numero_cluster= 4 (3.0/2.1)
- Se ((numero_BIPs= 2) & (Media_HR > 87.69)) allora numero_cluster= 5 (3.0/2.1)
- Se (numero_BIPs= 3) allora numero_cluster= 4 (7.0/3.4)
- Se ((4 <= numero_BIPs <= 5) & (texture= alta) & (Media_HR <= 87.69)) allora numero_cluster= 1
(2.0/1.8)
- Se ((4 <= numero_BIPs <= 5) & (texture= alta) & (87.69 < Media_HR <= 89.63)) allora
numero_cluster= 2 (2.0/1.0)
- Se ((4 <= numero_BIPs <= 5) & (texture= alta) & (Media_HR > 89.63)) allora numero_cluster= 5
(3.0/1.1)
- Se ((4 <= numero_BIPs <= 5) & (texture= bassa)) allora numero_cluster= 4 (3.0/2.1)
- Se ((6 <= numero_BIPs <= 16) & (texture= alta)) allora numero_cluster= 0 (2.0/1.8)
- Se ((6 <= numero_BIPs <= 16) & (texture= bassa)) allora numero_cluster= 5 (5.0/2.3)

L)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((Age= 19) & (Media_HR <= 87.69)) allora numero_cluster= 0 (2.0/1.8)
- Se ((Age= 19) & (Media_HR > 87.69)) allora numero_cluster= 3 (2.0/1.0)
- Se (Age= 20) allora numero_cluster= 2 (3.0/2.8)
- Se (Age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (Age= 22) allora numero_cluster= 0 (1.0/0.8)

- Se (Age= 23) allora numero_cluster= 2 (2.0/1.8)
- Se (Age= 24) allora numero_cluster= 1 (4.0/1.2)
- Se (Age= 25) allora numero_cluster= 2 (3.0/2.1)
- Se (Age= 26) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 27) allora numero_cluster= 0 (2.0/1.8)
- Se (Age= 28) allora numero_cluster= 1 (3.0/2.8)
- Se (Age= 29) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 30) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 32) allora numero_cluster= 4 (1.0/0.8)
- Se (Age= 34) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 35) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 40) allora numero_cluster= 4 (1.0/0.8)
- Se (Age= 41) allora numero_cluster= 4 (1.0/0.8)
- Se (Age= 43) allora numero_cluster= 1 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((Age= 19) & (Media_HR <= 87.69)) allora numero_cluster= 0 (2.0/1.8)
- Se ((Age= 19) & (Media_HR > 87.69)) allora numero_cluster= 5 (2.0/1.0)
- Se (Age= 20) allora numero_cluster= 2 (3.0/2.8)
- Se (Age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (Age= 22) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 23) allora numero_cluster= 3 (2.0/1.8)
- Se (Age= 24) allora numero_cluster= 1 (4.0/1.2)
- Se (Age= 25) allora numero_cluster= 4 (3.0/2.1)
- Se (Age= 26) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 27) allora numero_cluster= 0 (2.0/1.8)
- Se (Age= 28) allora numero_cluster= 1 (3.0/2.8)
- Se (Age= 29) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 30) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 32) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 34) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 35) allora numero_cluster= 5 (1.0/0.8)
- Se (Age= 40) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 41) allora numero_cluster= 2 (1.0/0.8)
- Se (Age= 43) allora numero_cluster= 1 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- (numero_cluster= 5) (40.0/29.4)

M)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((Age= 19) & (Media_HR <= 87.69)) allora numero_cluster= 0 (2.0/1.8)
- Se ((Age= 19) & (Media_HR > 87.69)) allora numero_cluster= 3 (2.0/1.0)
- Se (Age= 20) allora numero_cluster= 2 (3.0/2.8)
- Se (Age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (Age= 22) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 23) allora numero_cluster= 2 (2.0/1.8)
- Se (Age= 24) allora numero_cluster= 1 (4.0/1.2)
- Se (Age= 25) allora numero_cluster= 2 (3.0/2.1)
- Se (Age= 26) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 27) allora numero_cluster= 0 (2.0/1.8)
- Se (Age= 28) allora numero_cluster= 1 (3.0/2.8)
- Se (Age= 29) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 30) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 32) allora numero_cluster= 4 (1.0/0.8)
- Se (Age= 34) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 35) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 40) allora numero_cluster= 4 (1.0/0.8)
- Se (Age= 41) allora numero_cluster= 4 (1.0/0.8)
- Se (Age= 43) allora numero_cluster= 1 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((Age= 19) & (Media_HR <= 87.69)) allora numero_cluster= 0 (2.0/1.8)
- Se ((Age= 19) & (Media_HR > 87.69)) allora numero_cluster= 5 (2.0/1.0)
- Se (Age= 20) allora numero_cluster= 2 (3.0/2.8)
- Se (Age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (Age= 22) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 23) allora numero_cluster= 3 (2.0/1.8)

- Se (Age= 24) allora numero_cluster= 1 (4.0/1.2)
- Se (Age= 25) allora numero_cluster= 4 (3.0/2.1)
- Se (Age= 26) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 27) allora numero_cluster= 0 (2.0/1.8)
- Se (Age= 28) allora numero_cluster= 1 (3.0/2.8)
- Se (Age= 29) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 30) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 32) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 34) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 35) allora numero_cluster= 5 (1.0/0.8)
- Se (Age= 40) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 41) allora numero_cluster= 2 (1.0/0.8)
- Se (Age= 43) allora numero_cluster= 1 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- (numero_cluster= 5) (40.0/29.4)

N)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((Age= 19) & (Media_HR <= 87.69)) allora numero_cluster= 0 (2.0/1.8)
- Se ((Age= 19) & (Media_HR > 87.69)) allora numero_cluster= 3 (2.0/1.0)
- Se (Age= 20) allora numero_cluster= 2 (3.0/2.8)
- Se (Age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (Age= 22) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 23) allora numero_cluster= 2 (2.0/1.8)
- Se (Age= 24) allora numero_cluster= 1 (4.0/1.2)
- Se (Age= 25) allora numero_cluster= 2 (3.0/2.1)
- Se (Age= 26) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 27) allora numero_cluster= 0 (2.0/1.8)
- Se (Age= 28) allora numero_cluster= 1 (3.0/2.8)
- Se (Age= 29) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 30) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 32) allora numero_cluster= 4 (1.0/0.8)

- Se (Age= 34) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 35) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 40) allora numero_cluster= 4 (1.0/0.8)
- Se (Age= 41) allora numero_cluster= 4 (1.0/0.8)
- Se (Age= 43) allora numero_cluster= 1 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Age= 18) allora numero_cluster= 0 (3.0/2.1)
- Se ((Age= 19) & (Media_HR <= 87.69)) allora numero_cluster= 0 (2.0/1.8)
- Se ((Age= 19) & (Media_HR > 87.69)) allora numero_cluster= 5 (2.0/1.0)
- Se (Age= 20) allora numero_cluster= 2 (3.0/2.8)
- Se (Age= 21) allora numero_cluster= 0 (3.0/2.8)
- Se (Age= 22) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 23) allora numero_cluster= 3 (2.0/1.8)
- Se (Age= 24) allora numero_cluster= 1 (4.0/1.2)
- Se (Age= 25) allora numero_cluster= 4 (3.0/2.1)
- Se (Age= 26) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 27) allora numero_cluster= 0 (2.0/1.8)
- Se (Age= 28) allora numero_cluster= 1 (3.0/2.8)
- Se (Age= 29) allora numero_cluster= 0 (1.0/0.8)
- Se (Age= 30) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 32) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 34) allora numero_cluster= 1 (2.0/1.8)
- Se (Age= 35) allora numero_cluster= 5 (1.0/0.8)
- Se (Age= 40) allora numero_cluster= 3 (1.0/0.8)
- Se (Age= 41) allora numero_cluster= 2 (1.0/0.8)
- Se (Age= 43) allora numero_cluster= 1 (1.0/0.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (numero_cluster= 4) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- (numero_cluster= 5) (40.0/29.4)

O)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- numero_BIPs= 2 (40.0/34.1)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (numero_cluster= 0) allora numero_BIPs= 0 (7.0/6.1)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (10.0/9.1)
- Se (numero_cluster= 2) allora numero_BIPs= 0 (5.0/4.1)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (5.0/4.1)
- Se (numero_cluster= 4) allora numero_BIPs= 2 (5.0/4.1)
- Se (numero_cluster= 5) allora numero_BIPs= 5 (8.0/6.3)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- Se (numero_cluster= 0) allora numero_BIPs= 16 (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)
- Se (numero_cluster= 2) allora numero_BIPs= 0 (5.0/4.1)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (9.0/6.4)
- Se (numero_cluster= 4) allora numero_BIPs= 3 (23.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se (numero_cluster= 0) allora numero_BIPs= 16 (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)
- Se (numero_cluster= 2) allora numero_BIPs= 0 (5.0/4.1)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (9.0/6.4)
- Se (numero_cluster= 4) allora numero_BIPs= 3 (10.0/6.5)
- Se (numero_cluster= 5) allora numero_BIPs= 0 (13.0/12.1)

P)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se ((numero_cluster= 0) & (sesso= femmina)) allora numero_BIPs= 5 (4.0/3.1)
- Se ((numero_cluster= 0) & (sesso= maschio)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= alta)) allora numero_BIPs= 4 (4.0/3.1)
- Se ((numero_cluster= 1) & (texture= bassa) & (avatar= cartoon)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= bassa) & (avatar= realistico)) allora numero_BIPs= 3 (3.0/2.8)
- Se ((numero_cluster= 2) & (texture= alta)) allora numero_BIPs= 1 (3.0/2.8)
- Se ((numero_cluster= 2) & (texture= bassa)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta)) allora numero_BIPs= 5 (6.0/4.3)
- Se ((numero_cluster= 3) & (texture= bassa)) allora numero_BIPs= 2 (2.0/1.8)

- Se ((numero_cluster= 4) & (texture= alta) & (sesso= femmina)) allora numero_BIPs= 0(3.0/2.1)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= maschio)) allora numero_BIPs= 2(2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa)) allora numero_BIPs= 3 (5.0/4.1)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se ((numero_cluster= 0) & (sesso= femmina)) allora numero_BIPs= 5 (4.0/3.1)
- Se ((numero_cluster= 0) & (sesso= maschio)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= alta)) allora numero_BIPs= 4 (4.0/3.1)
- Se ((numero_cluster= 1) & (texture= bassa) & (avatar= cartoon)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= bassa) & (avatar= realistico)) allora numero_BIPs= 3 (3.0/2.8)
- Se ((numero_cluster= 2) & (texture= alta)) allora numero_BIPs= 0 (3.0/2.1)
- Se ((numero_cluster= 2) & (texture= bassa)) allora numero_BIPs= 3 (2.0/1.0)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (5.0/4.1)
- Se ((numero_cluster= 4) & (texture= alta)) allora numero_BIPs= 1 (3.0/2.8)
- Se ((numero_cluster= 4) & (texture= bassa)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta)) allora numero_BIPs= 5 (6.0/4.3)
- Se ((numero_cluster= 5) & (texture= bassa)) allora numero_BIPs= 2 (2.0/1.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- Se (numero_cluster= 0) allora numero_BIPs= 16 (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= cartoon)) allora numero_BIPs= 4 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= realistico)) allora numero_BIPs= 0 (3.0/2.1)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (9.0/6.4)
- Se ((numero_cluster= 4) & (sesso= femmina)) allora numero_BIPs= 5 (11.0/8.5)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= alta)) allora numero_BIPs= 1(4.0/3.1)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa)) allora numero_BIPs= 3 (8.0/6.3)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se (numero_cluster= 0) allora numero_BIPs= 16 (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= cartoon)) allora numero_BIPs= 4 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= realistico)) allora numero_BIPs= 0 (3.0/2.1)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (9.0/6.4)

- Se ((numero_cluster= 4) & (sesso= maschio)) allora numero_BIPs= 3 (5.0/3.2)
- Se ((numero_cluster= 4) & (sesso= femmina) & (texture= alta)) allora numero_BIPs= 3(3.0/2.1)
- Se ((numero_cluster= 4) & (sesso= femmina) & (texture= bassa)) allora numero_BIPs= 5 (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (sesso= femmina)) allora numero_BIPs= 5(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (sesso= maschio)) allora numero_BIPs= 1(3.0/2.1)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= femmina) & (avatar= cartoon)) allora numero_BIPs= 0 (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= femmina) & (avatar= realistico)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= maschio) & (avatar= cartoon)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= maschio) & (avatar= realistico)) allora numero_BIPs= 6 (2.0/1.8)

Q)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se ((numero_cluster= 0) & (sesso= femmina)) allora numero_BIPs= 5 (4.0/3.1)
- Se ((numero_cluster= 0) & (sesso= maschio)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= alta) & (Media_HR <= 95.19)) allora numero_BIPs= 4 (2.0/1.0)
- Se ((numero_cluster= 1) & (texture= alta) & (Media_HR > 95.19)) allora numero_BIPs= 1 (2.0/1.8)
- Se ((numero_cluster= 1) & (texture= bassa) & (avatar= cartoon)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= bassa) & (avatar= realistico)) allora numero_BIPs= 3 (3.0/2.8)
- Se ((numero_cluster= 2) & (texture= alta)) allora numero_BIPs= 1 (3.0/2.8)
- Se ((numero_cluster= 2) & (texture= bassa)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 3) & (texture= alta) & (Media_HR <= 86.8)) allora numero_BIPs= 1 (3.0/2.8)
- Se ((numero_cluster= 3) & (texture= alta) & (Media_HR >86.8)) allora numero_BIPs= 5 (3.0/1.1)
- Se ((numero_cluster= 3) & (texture= bassa)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= femmina)) allora numero_BIPs= 0 (3.0/2.1)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= maschio)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa)) allora numero_BIPs= 3 (5.0/4.1)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se ((numero_cluster= 0) & (sesso= femmina)) allora numero_BIPs= 5 (4.0/3.1)
- Se ((numero_cluster= 0) & (sesso= maschio)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= alta) & (Media_HR <= 95.19)) allora numero_BIPs= 4 (2.0/1.0)
- Se ((numero_cluster= 1) & (texture= alta) & (Media_HR > 95.19)) allora numero_BIPs= 1 (2.0/1.8)
- Se ((numero_cluster= 1) & (texture= bassa) & (avatar= cartoon)) allora numero_BIPs= 0 (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= bassa) & (avatar= realistico)) allora numero_BIPs= 3 (3.0/2.8)
- Se ((numero_cluster= 2) & (texture= alta)) allora numero_BIPs= 0 (3.0/2.1)
- Se ((numero_cluster= 2) & (texture= bassa)) allora numero_BIPs= 3 (2.0/1.0)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (5.0/ 4.1)
- Se ((numero_cluster= 4) & (texture= alta)) allora numero_BIPs= 1 (3.0/2.8)
- Se ((numero_cluster= 4) & (texture= bassa)) allora numero_BIPs= 2 (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (Media_HR <= 86.8)) allora numero_BIPs= 1 (3.0/2.8)
- Se ((numero_cluster= 5) & (texture= alta) & (Media_HR >86.8)) allora numero_BIPs= 5 (3.0/1.1)
- Se ((numero_cluster= 5) & (texture= bassa)) allora numero_BIPs= 2 (2.0/1.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- Se (numero_cluster= 0) allora numero_BIPs= 16 (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= cartoon)) allora numero_BIPs= 4 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= realistico)) allora numero_BIPs= 0 (3.0/2.1)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (9.0/6.4)
- Se ((numero_cluster= 4) & (sesso= femmina)) allora numero_BIPs= 5 (11.0/8.5)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= alta)) allora numero_BIPs= 1 (4.0/3.1)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa) & (Media_HR <= 89.95)) allora numero_BIPs= 3 (5.0/3.2)
- Se ((numero_cluster= 4) & (sesso= maschio) & (texture= bassa) & (Media_HR > 89.95)) allora numero_BIPs= 2 (3.0/2.8)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se (numero_cluster= 0) allora numero_BIPs= 16 (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)

- Se ((numero_cluster= 2) & (avatar= cartoon)) allora numero_BIPs= 4 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= realistico)) allora numero_BIPs= 0 (3.0/2.1)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (9.0/6.4)
- Se ((numero_cluster= 4) & (sesso= maschio)) allora numero_BIPs= 3 (5.0/3.2)
- Se ((numero_cluster= 4) & (sesso= femmina) & (texture= alta)) allora numero_BIPs= 3(3.0/2.1)
- Se ((numero_cluster= 4) & (sesso= femmina) & (texture= bassa)) allora numero_BIPs= 5 (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (sesso= femmina)) allora numero_BIPs= 5(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (sesso= maschio)) allora numero_BIPs= 1 (3.0/2.1)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= femmina) & (avatar= cartoon)) allora numero_BIPs= 0 (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= femmina) & (avatar= realistico)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= maschio) & (Media_HR <= 90.9)) allora numero_BIPs= 6 (2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= maschio) & (Media_HR > 90.9)) allora numero_BIPs= 2 (2.0/1.8)

R)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- $0 \leq \text{numero_BIPs} \leq 3$ (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- $0 \leq \text{numero_BIPs} \leq 3$ (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- $(0 \leq \text{numero_BIPs} \leq 3)$ (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- $(0 \leq \text{numero_BIPs} \leq 1)$ (40.0/19.7)

S)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se $(\text{Media_HR} \leq 84.81) \& (\text{texture} = \text{alta})$ allora $(0 \leq \text{numero_BIPs} \leq 3)$ (6.0/2.3)
- Se $(84.81 \leq \text{Media_HR} \leq 93.13) \& (\text{texture} = \text{alta})$ allora $(4 \leq \text{numero_BIPs} \leq 7)$ (6.0/1.2)
- Se $(\text{Media_HR} \leq 93.13) \& (\text{texture} = \text{bassa})$ allora $(0 \leq \text{numero_BIPs} \leq 3)$ (16.0/7.9)

- Se (Media_HR > 93.13) allora (0 <= numero_BIPs <= 3) (12.0/5.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- 0 <= numero_BIPs <= 3 (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (0 <= numero_BIPs <= 3) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- (0 <= numero_BIPs <= 3) (40.0/19.7)

T)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se ((texture= alta) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 5) (10.0/7.5)
- Se ((texture= alta) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (10.0/6.5)
- Se ((texture= bassa) & (numero_cluster= 0) & (sesso= femmina)) allora (4 <= numero_BIPs <= 5) (3.0/2.1)
- Se ((texture= bassa) & (numero_cluster= 0) & (sesso= maschio)) allora (numero_BIPs= 2) (2.0/1.8)
- Se ((texture= bassa) & (numero_cluster= 1)) allora (6 <= numero_BIPs <= 16) (6.0/4.3)
- Se ((texture= bassa) & (numero_cluster= 2)) allora (numero_BIPs= 2) (2.0/1.0)
- Se ((texture= bassa) & (numero_cluster= 3)) allora (numero_BIPs= 2) (2.0/1.8)
- Se ((texture= bassa) & (numero_cluster= 4)) allora (numero_BIPs= 3) (5.0/4.1)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se ((numero_cluster= 0) & (sesso= femmina)) allora (4 <= numero_BIPs <= 5) (4.0/3.1)
- Se ((numero_cluster= 0) & (sesso= maschio)) allora (0 <= numero_BIPs <= 1) (3.0/2.8)
- Se ((numero_cluster= 1) & (texture= alta)) allora (4 <= numero_BIPs <= 5) (4.0/3.1)
- Se ((numero_cluster= 1) & (texture= bassa)) allora (numero_BIPs= 16) (6.0/4.3)
- Se ((numero_cluster= 2) & (texture= alta)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se ((numero_cluster= 2) & (texture= bassa)) allora (numero_BIPs= 3) (2.0/1.0)
- Se ((numero_cluster= 3) & (sesso= femmina)) allora (numero_BIPs= 2) (2.0/1.8)
- Se ((numero_cluster= 3) & (sesso= maschio)) allora (6 <= numero_BIPs <= 16) (3.0/2.1)
- Se ((numero_cluster= 4) & (texture= alta)) allora (0 <= numero_BIPs <= 1) (3.0/2.8)
- Se ((numero_cluster= 4) & (texture= bassa)) allora (numero_BIPs= 2) (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta)) allora (4 <= numero_BIPs <= 5) (6.0/4.3)

- Se ((numero_cluster= 5) & (texture= bassa) allora (numero_BIPs= 2) (2.0/1.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- Se (texture= alta) allora (4 <= numero_BIPs <= 5) (20.0/14.9)
- Se ((texture= bassa) & (sesso= femmina) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 1) (5.0/3.2)
- Se ((texture= bassa) & (sesso= femmina) & (avatar= realistico)) allora (numero_BIPs= 2) (5.0/3.2)
- Se ((texture= bassa) & (sesso= maschio)) allora (numero_BIPs= 3) (10.0/7.5)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se (numero_cluster= 0) allora (6 <= numero_BIPs <= 16) (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 2) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se (numero_cluster= 3) allora numero_BIPs= 2 (9.0/6.4)
- Se ((numero_cluster= 4) & (sesso= femmina) & (texture= alta)) allora numero_BIPs= 3 (3.0/2.1)
- Se ((numero_cluster= 4) & (sesso= femmina) & (texture= bassa)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 4) & (sesso= maschio)) allora numero_BIPs= 3 (5.0/3.2)
- Se ((numero_cluster= 5) & (texture= alta) & (sesso= femmina)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (sesso= maschio)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= femmina) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 1) (2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= femmina) & (avatar= realistico)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= maschio)) allora (6 <= numero_BIPs <= 16) (4.0/2.2)

U)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se ((texture= alta) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 5) (10.0/7.5)
- Se ((texture= alta) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (10.0/6.5)
- Se ((texture= bassa) & (sesso= femmina) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 5) (5.0/3.2)

- Se ((texture= bassa) & (sesso= femmina) & (avatar= realistico)) allora (numero_BIPs= 2) (5.0/3.2)
- Se ((texture= bassa) & (sesso= maschio)) allora (6 <= numero_BIPs<=16) (10.0/7.5)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se ((texture= alta) & (Media_HR <= 95.19)) allora (4 <= numero_BIPs <= 5) (13.0/7.7)
- Se ((texture= alta) & (Media_HR > 95.19)) allora (0<= numero_BIPs <= 1) (7.0/5.3)
- Se (texture= bassa) allora (6 <= numero_BIPs <= 16) (20.0/15.8)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- Se (numero_cluster= 0) allora (6 <= numero_BIPs <= 16) (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= cartoon)) allora (4 <= numero_BIPs <= 5) (2.0/1.0)
- Se ((numero_cluster= 2) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se ((numero_cluster= 3) & (Media_HR <= 84.81)) allora (4 <= numero_BIPs <= 5) (6.0/4.3)
- Se ((numero_cluster= 3) & (Media_HR > 84.81)) allora (numero_BIPs= 2) (3.0/2.1)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= femmina) & (Media_HR <= 84.81)) allora numero_BIPs= 2 (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= femmina) & (Media_HR > 84.81)) allora (4 <=numero_BIPs<= 5) (3.0/2.1)
- Se ((numero_cluster= 4) & (texture= alta) & (sesso= maschio)) allora (0 <= numero_BIPs <=1) (4.0/2.2)
- Se ((numero_cluster= 4) & (texture= bassa) & (sesso= femmina) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 1) (4.0/3.1)
- Se ((numero_cluster= 4) & (texture= bassa) & (sesso= femmina) & (avatar= realistico)) allora (numero_BIPs= 2) (2.0/1.8)
- Se ((numero_cluster= 4) & (texture= bassa) & (sesso= maschio) & (Media_HR <= 89.95)) allora numero_BIPs= 3 (5.0/3.2)
- Se ((numero_cluster= 4) & (texture= bassa) & (sesso= maschio) & (Media_HR > 89.95)) allora (6 <= numero_BIPs<= 16) (3.0/2.1)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se ((numero_cluster= 0) allora (6 <= numero_BIPs <= 16) (1.0/0.8)
- Se (numero_cluster= 1) allora numero_BIPs= 3 (2.0/1.8)
- Se ((numero_cluster= 2) & (avatar= cartoon)) allora (4 <= numero_BIPs <=5) (2.0/1.0)
- Se ((numero_cluster= 2) & (avatar= realistico)) allora (0 <= numero_BIPs <= 1) (3.0/2.1)
- Se ((numero_cluster= 3) & (Media_HR <= 84.81)) allora (4 <= numero_BIPs <= 5) (6.0/4.3)
- Se ((numero_cluster= 3) & (Media_HR > 84.81)) allora (numero_BIPs= 2) (3.0/2.1)

- Se ((numero_cluster= 4) & (sesso= femmina) & (texture= alta)) allora numero_BIPs= 3(3.0/2.1)
- Se ((numero_cluster= 4) & (sesso= femmina) & (texture= bassa)) allora (4 <= numero_BIPs <= 5)
(2.0/1.0)
- Se ((numero_cluster= 4) & (sesso= maschio)) allora numero_BIPs= 3 (5.0/3.2)
- Se ((numero_cluster= 5) & (texture= alta) & (sesso= femmina)) allora (4 <= numero_BIPs <= 5)
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= alta) & (sesso= maschio)) allora (0 <= numero_BIPs <= 1)
(3.0/2.1)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= femmina) & (avatar= cartoon)) allora (0 <= numero_BIPs <= 1)
(2.0/1.0)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= femmina) & (avatar= realistico)) allora
(numero_BIPs= 2) (2.0/1.8)
- Se ((numero_cluster= 5) & (texture= bassa) & (sesso= maschio)) allora (6 <= numero_BIPs <= 16)
(4.0/2.2)

V)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Media_HR <= 95.19) allora numero_BIPs= 3 (31.0/25.9)
- Se (Media_HR > 95.19) allora numero_BIPs= 0 (9.0/7.3)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Media_HR <= 95.19) allora numero_BIPs= 3 (31.0/25.9)
- Se (Media_HR > 95.19) allora numero_BIPs= 0 (9.0/7.3)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- Se (Media_HR <= 95.19) allora numero_BIPs= 3 (31.0/25.9)
- Se (Media_HR > 95.19) allora numero_BIPs= 0 (9.0/7.3)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- Se (Media_HR <= 95.19) allora numero_BIPs= 3 (31.0/25.9)
- Se (Media_HR > 95.19) allora numero_BIPs= 0 (9.0/7.3)

Z)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- (0 <=numero_BIPs<=3) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- (0 <=numero_BIPs<=3) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- (0 <= numero_BIPs <= 3) (40.0/19.7)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*

- (0 <=numero_BIPs<=3) (40.0/19.7)

AA)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione min-max*

- Se (Media_HR <= 95.19) allora (numero_BIPs = 3) (31.0/23.2)
- Se (Media_HR > 95.19) allora (0<= numero_BIPs <= 1) (9.0/6.4)

➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione min-max*

- Se (Media_HR <= 95.19) allora (numero_BIPs = 3) (31.0/23.2)
- Se (Media_HR > 95.19) allora (0<= numero_BIPs <= 1) (9.0/6.4)

➤ *Caso: Clusters determinati con gerarchico, k= 5 e normalizzazione z-score*

- Se (Media_HR <= 95.19) allora numero_BIPs= 3 (31.0/23.2)
- Se (Media_HR > 95.19) allora (0 <= numero_BIPs <= 1) (9.0/6.4)


➤ *Caso: Clusters determinati con gerarchico, k= 6 e normalizzazione z-score*


- Se (Media_HR <= 95.19) allora numero_BIPs= 3 (31.0/23.2)
- Se (Media_HR > 95.19) allora (0 <= numero_BIPs <= 1) (9.0/6.4)

		A	B	C	D	E	F	G	H	I	L	M	N	O	P	Q	R	S	T	U	V	Z	AA
<i>clustering</i>																							
<i>Bips</i>	<i>Valori</i>																						
	<i>Int.regolari</i>																						
	<i>Int.irregolari</i>																						
<i>Media Heart Rate</i>																							
<i>Texture</i>																							
<i>Avatar</i>																							
<i>Sesso</i>																							
<i>Altri Dati(*)</i>																							

Tabella D-2 Attributi utilizzati nelle varie classificazioni

Legenda

 target

 attributo

(*) AltriDati: Occupy/ Medicine/ Language/ Alcohol/ Literate/ Programming/ VReality/ Games/ Gamesimes

Indice delle figure

<i>Figura 2.1</i>	<i>Fasi del KDD</i>	15
<i>Figura 2.2</i>	<i>Architettura di un sistema KDD</i>	16
<i>Figura 2.3</i>	<i>Ricerca di serie temporali ricorrenti</i>	21
<i>Figura 3.1</i>	<i>Tabella di contingenza per variabili binarie</i>	33
<i>Figure 3.2</i>	<i>Pseudocodice dell'algoritmo k-means</i>	39
<i>Figura 3.3</i>	<i>Esempio di clustering realizzato con il k-means</i>	40
<i>Figura 3.4</i>	<i>Esempio di clustering realizzato con algoritmo gerarchico</i>	43
<i>Figura 3.5</i>	<i>Non transitività della nozione di similarità per LCSS</i>	45
<i>Figura 3.6</i>	<i>Classificazione: training set</i>	46
<i>Figura 3.7</i>	<i>Pseudocodice di un algoritmo per la generazione di un albero di decisione</i>	51
<i>Figura 3.8</i>	<i>Esempio di classificazione mediante un albero di decisione</i>	53
<i>Figura 4.1 (a)</i>	<i>Il primo HMD</i>	67
<i>Figura 4.1 (b)</i>	<i>HMD moderno</i>	67
<i>Figura 4.2</i>	<i>CAVE</i>	68
<i>Figura 4.3</i>	<i>Manipolazione di oggetti: data gloves</i>	69
<i>Figura 4.4</i>	<i>Applicazione della realtà virtuale ai beni culturali</i>	75
<i>Figura 4.5</i>	<i>Applicazione della realtà virtuale in campo medico</i>	75
<i>Figura 4.6</i>	<i>Utilizzo della realtà virtuale per gli studi architettonici</i>	76
<i>Figura 5.1</i>	<i>Centro commerciale "virtuale"</i>	79
<i>Figura 5.2 (a)</i>	<i>Higher Fidelity Characters</i>	81
<i>Figura 5.2 (b)</i>	<i>Cartoon Form Characters</i>	81
<i>Figura 6.1</i>	<i>Organizzazione del file contenente le coordinate spaziali lungo l'asse x (rispett. asse z)</i>	96
<i>Figura 6.2</i>	<i>Organizzazione dei dati nel file di testo fornito in input</i>	

	<i>all'algoritmo di clustering</i>	97
<i>Figura 6.3</i>	<i>Rappresentazione delle traiettorie nel centro commerciale virtuale</i>	99
<i>Figura 6.4</i>	<i>Rappresentazione delle traiettorie medie nel caso $k=5$</i>	105
<i>Figura 6.5</i>	<i>Rappresentazione delle traiettorie medie nel caso $k=6$</i>	106
<i>Figura 6.6</i>	<i>Rappresentazione delle traiettorie appartenenti al cluster 0</i>	107
<i>Figura 6.7</i>	<i>Rappresentazione delle traiettorie appartenenti al cluster 3</i>	108
<i>Figura 6.8</i>	<i>Albero di classificazione</i>	113
<i>Figura 6.9</i>	<i>Organizzazione del file di input per l'algoritmo di clustering nell'approccio alternativo</i>	132
<i>Figura A-1</i>	<i>Mapping del centro commerciale virtuale su un piano bidimensionale</i>	137
<i>Figura A-2</i>	<i>Frammento del file pos_x</i>	139
<i>Figura A-3</i>	<i>Frammento del file timing</i>	140
<i>Figura A-4</i>	<i>Frammento del file bips</i>	141

Indice delle tabelle

<i>Tabella 5.1</i>	<i>Caratteristiche degli esperimenti condotti</i>	<i>78</i>
<i>Tabella A-1</i>	<i>Caratteristiche dell'ambiente virtuale per ciascun soggetto</i>	<i>143</i>
<i>Tabella A-2</i>	<i>Dati demografici dei partecipanti all'esperimento</i>	<i>147</i>
<i>Tabella A-3</i>	<i>Conoscenze informatiche e di realtà virtuale di ciascun partecipante</i>	<i>148</i>
<i>Tabella B-1</i>	<i>Suddivisione delle traiettorie in clusters</i>	<i>150</i>
<i>Tabella B-2</i>	<i>Attributi utilizzati nelle varie classificazioni</i>	<i>221</i>
<i>Tabella B-3</i>	<i>Approccio clusterNotcluster: attributi considerati</i>	<i>222</i>
<i>Tabella C-1</i>	<i>Attributi utilizzati nelle varie classificazioni</i>	<i>253</i>
<i>Tabella D-1</i>	<i>Suddivisione delle traiettorie in clusters</i>	<i>256</i>
<i>Tabella D-2</i>	<i>Attributi utilizzati nelle varie classificazioni</i>	<i>281</i>

Bibliografia

- [And00] Andreassi J., *Psychophysiology: Human Behavior and Physiological Response*, 2000, Lawrence Erlbaum Associates, London, UK, 4th edition.
- [Ber97] Michael J.A. Berry, Gordon Linoff, *Data Mining Techniques*, 1997, Wiley Computer Publishing.
- [Bro03] Brogni A., Slater M., and Steed A., *More breaks less presence*, 2003, In Presence 2003: the 6th Annual International Workshop on Presence, Alicante, Spain, Presence-research.
- [Broo99] Brooks F.P., *What's real about Virtual Reality*, 1999, Special Report, pages 16-27.
- [Bur94] Burdea G., Coiffet P., *Virtual Reality Technology*, 1994, Wiley.
- [Car97] Carletti A., Tarantini A. Varani A., *Realtà virtuale ed apprendimento*, 1997.
- [Cla92] Clarian R.B., *Media Research with a Galvanic Skin Response Biosensor*, 1992, Presentation at the annual Convention of the association for Educational Communication and Technology Washington, DC.
- [Cru93] Cruz-Neira C, Sandin D.J., Defanti T.A., *Surround-Screen Projection-Based Virtual Reality: the Design and Implementation of the CAVE*, 1993, Proceedings of Computer Graphics (SIGGRAPH), pp. 135-142.

- [Fra91] W.J. Frawley, G. Piatetski-Shapiro y C.J. Matheus, *Knowledge Discovery in databases: An Overview*, Knowledge Discovery In Databases, 1991, G. Piatetski-Shapiro y W. Frawley, AAAI-MIT Press, Menlo Park, California, pag 1-27.
- [Fre00] Freeman J., Avons Pearson D.E., Meddis R., and IJsselsteijn W.A., *Using behavioural realism to estimate presence: A study of the utility of postural responses to motion-stimuli*, 2000, Presence: Teleoperators and Virtual Environments, 9(2): 149-164.
- [Fre01] Freeman J., Lessiter J., and IJsselsteijn W.A., *An introduction to presence: A sense of being there in a mediated environment*, 2001, The Psychologist, British Psychological Society.
- [Gag03] Gaggioli A., Bassi M., and Fave A.D., *Quality of Experience in Virtual Environments*, chapter 8, pages 121-135, 2003, IOS Press, Nieuwe Hemweg 6B, 1013 BG Amsterdam, The Netherlands.
- [Gar03] Garau M., Slater M., Vinayagamoorthy, Brogni A., Steed A., Sasse M.A., *The Impact of Avatar Realism and eye gaze control on perceived quality of communication in a Shared Immersive Virtual Environment*, 2003, Paper: New Directions in Video Conferencing, 529-536.
- [Gug04] Guger C., Edlinger G., Leeb R., Pfurtscheller G., Antley A., Garau M., Brogni A., Friedman D., and Slater M., *Heart rate variability and event-related ecg in virtual environments*, 2004, In Presence 2004: The 7th Annual International Workshop on Presence.

- [Han00] Han J., and Kamber M., *Data Mining: Concepts and Technique Techniques*, 2000, San Francisco, CA: Morgan Kaufmann
- [Hand01] Hand D., Mannila H., Smyth P., *Principles of Data Mining*, 2001, A Bradford book The MIT Press, Cambridge, Massachusetts London England.
- [IJ00] IJsselsteijn W.A., de Ridder H., Freeman J., and Avons S.E., *Presence: Concept, determinants and measurement*, 2000, In Proceedings of the SPIE, Human Vision and Electronic Imaging, pages 3959 – 3976.
- [IJ01] IJsselsteijn W.A., Freeman J. and Huib de Ridder, Editorial: *Presence: Where Are We?*, 2001, Cyberpsychology and Behavior, 4(2) 179-182.
- [Ins03] Insko B.E., *Measuring Presence: Subjective, Behavioral And Physiological Methods*, 2003, chapter 7, pages 109-119. IOS Press, Nieuwe Hemweg 6B, 1013 BG Amsterdam, The Netherlands.
- [Ist98] Istale J., *What is Virtual Reality*, 1998, Virtual reality Information resources.
- [Lar05] Larose D.T., *Discovering Knowledge in Data, an introduction to data mining*, 2005, Wiley-interscience.
- [Lis04] Lisetti C.L., Nasoz F., *Using Noninvasive Wearable Computers to recognize human emotions from physiological signals*, 2004, Journal on Applied Signal Processing, 1672-1687.

- [Lom97] Lombard M. and Ditton T.B., *At the heart of it all: The Concept of Presence*, 1997, Journal of Computer-Mediated Communication, 3(2):225-240.
- [Man00] Manganaro, *Introduzione al KDD e al data mining*, 2000.
- [Man01] Manganaro, *Tecniche di DM: Alberi di decisione ed algoritmi di classificazione*.
- [Mee00] Meehan M., *An objective surrogate for presence: Physiological response*, 2000, In PRESENCE 2000 - 3rd International Workshop on Presence, Delft, Netherlands.
- [Mee02] Meehan M., Insko B., Whitton M., and Frederick P., Brooks J., *Physiological measures of presence in stressful virtual environments*, 2002, In Proceedings of ACM – Transactions on Graphics, volume 21, pages 645-653. ACM – SIGGRAPH.
- [Nan02] Nanni M., *Clustering methods for spatio-temporal data*, 2002, PhD Thesis, Dipartimento di informatica, Università degli Studi di Pisa.
- [Non03] Nong Ye, *The handbook of data mining*, 2003, Arizona State University.
- [Riv03] Riva G., Davide F., and IJsselsteijn W.A., *Being There: Concepts, Effects and Measurements of User Presence In Synthetic Environments*, 2003, IOS Press, Nieuwe Hemweg 6B, 1013 BG Amsterdam, The Netherlands.
- [Rug04] Ruggieri S., *YaDT: Yet another Decision Tree builder*, 2004, Dipartimento di informatica, Università degli studi di Pisa.

- [San05] Sanchez-Vives M.V. and Slater M., *From presence to consciousness through virtual reality*, 2005, Nature Neuroscience, 6(4): 8-16.
- [She02] Sherman W.R. and Craig A.B., *Understanding Virtual Reality: interface, application, and design*, 2002, Morgan Kaufmann.
- [Sla94] Slater M., Usoh M. and Steed A., *Depth of Presence in Virtual Environments*, 1994, Presence: Teleoperators and Virtual Environments, 3.2, 130-144.
- [Sla96] Slater M., Linakis V., Usoh M. and Kooper R., *Immersion, presence, and performance in virtual environments: An experiment using tri-dimensional chess*, 1996, In Green M. editor, VRST 96: Proceedings of the ACM Symposium on Virtual Reality Software and Technology, pages 163-172, Hong Kong, ACM.
- [Sla97] Slater M. and Wilbur S., *A framework for immersive virtual environments (five): Speculations on the role of presence in virtual environments*, 1997, Presence: Teleoperators and Virtual Environments, 6 (6): 603-616.
- [Sla99] Slater M., *Measuring presence: A response to the witmer and singer questionnaire*, 1999, Presence: Teleoperators and Virtual Environments, 8(5): 560-566.
- [Sla00] Slater M. and Steed A., *A virtual presence counter*, 2000, Presence: Teleoperators and Virtual Environments, 9(5): 413-434.
- [Sla01] Slater M., *A Note on Presence Terminology*, 2001.

- [Sla02] Slater M., *Presence and the sixth sense*, 2002, Presence: Teleoperators and Virtual Environments, 11(4): 435-439.
- [Sla03] Slater M., Brogni A., and Steed A., *Physiological responses to breaks in Presence: A pilot study*, 2003, In Presence 2003: The 6th Annual International Workshop on Presence, Aalborg, Denmark.
- [TCC99] Two Crows Corporation, *Introduction to Data Mining and Knowledge Discovery*, 1999, third edition, data mining'99 Technology report.
- [Vic04] Viciano-Abad R., Reyes-Lecuona A., Garcia-Berdónes C., Diaz-Estrella A., Castillo- Carrion S., *The Importance of significant information in presence and stress within a virtual reality experience*, Dept. Tecnologia electrònica, University of Màlaga.
- [Vina03] Vinayagamoorthy V., Slater M., Steed A., *Emotional personification of humanoids in immersive virtual environments*, 2003, Department of Computer Science, University College London.
- [Vina04] Vinayagamoorthy V., Brogni A., Gillies M., Slater M., Steed A., *An Investigation of Presence Response across Variations in Visual Realism*, 2004.
- [Vin95] Vince J., *Virtual Reality Systems*, 1995, Addison Wesley, first edition.
- [Vlac02] Vlachos M., Kollios G., Gunopulos D., *Discovering Similar Multidimensional Trajectories*, 2002, In ICDE.

- [Vlac04] Vlachos M., Gunopulos D., Kollios G., *Robust Similarity Measures for Mobile Object Trajectories*, 2004.
- [Vre96] *Virtual reality: an overview* ([http:// www.ericdigest.org/1996-2/virtual.html](http://www.ericdigest.org/1996-2/virtual.html)).
- [Vrm] *Guida al VRML* from <http://www.vrml.it>.
- [Wit98] Witmer B.G. and Singer M. J., *Measuring presence in virtual environments: A presence questionnaire*, 1998, *Presence: Teleoperators and Virtual Environments*, 7(3): 225-240.